DOCUMENT RESUME

ED 248 847 IR 011 296

AUTHOR Torgerson, Shirley; And Others
TITLE LOGO in the Classroom.
INSTITUTION International Council for Computers in Education, Eugene, Oreg.
PUB DATE [84]
NOTE 220p.
AVAILABLE FROM International Council for Computers in Education, University of Oregon, 1787 Agate Street, Eugene, OR 97403-1923.
PUB TYPE Guides - Classroom Use - Guides (For Teachers) (052)

EDRS PRICE MF01 Plus Postage. PC Not Available from EDRS.
DESCRIPTORS Computer Assisted Instruction; *Computer Oriented Programs; *Computer Science Education; Elementary Education; Geometric Constructions; *Learning Activities; *Mathematics Instruction; Microcomputers; Objectives; *Programing; Programing Languages; Teaching Methods
IDENTIFIERS *LOGO Programing Language

ABSTRACT

This resource guide for teachers suggests activities for 20 instructional sessions for elementary children, particularly those in grades 3-6, to introduce them to the LOGO computer language. It can be used with five LOGO versions: Apple, Atari, MIT, Texas Instruments, and TRS. For each session, the guide covers objectives, concepts/skills, facilities/materials, and preparation; provides a chart of the relevant commands used in the five LOGO versions; describes student activities and assignments; and includes student worksheets. "Challenger" worksheets are provided for students who finish the basic assignments and are ready for more complex work. An introduction describes LOGO and discusses how it fits into the curriculum and how to get started. A scope and sequence chart indicates which of 10 educational goals are addressed by each session. Specific lesson topics include using the LOGO "primitives" as spatial commands, introducing students to the computer and to the LOGO turtle, changing directions by turning or pivoting, using a protractor, writing and editing procedures, rotation, LOGO procedures, recursion, and using variables as size input for geometric shapes. Fourteen charts for classroom use, answer keys, and a 17-item reference list are included. (LMM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED248847

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

- This document has been reproduced as received from the person or organization originating it.
  Minor changes have been made to improve reproduction quality.
- Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

Logo in the Classroom

"PERMISSION TO REPRODUCE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY

David Morsund

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

by
Shirley Torgerson
with assistance from
Mary Kay Kriley
& Janet Stone

ICCE Publications

IR011296

ACKNOWLEDGMENTS

WHEW! In print at last, thanks to the significant contributions of the ICCE staff. We are especially grateful to these people:

For professional editing and production assistance:

Sharon Franklin
Percy Franklin
Tim Volem

For gentle guidance in relevance and consistency, as well as impeccable attention to detail:

Anita Best

For improvements and comments, presented with the elegance of simplicity:

David Moursund

The atmosphere and attitude of the people at ICCE created an environment for learning and growing, a creative process in which we were able to share ideas and explore learning with Logo in the classroom. Thank you very much.

About the Authors

Shirley Torgerson teaches at Paxson Elementary School in Missoula, Montana. She has taught inservice classes and has given presentations on Logo and PILOT. As part of a pilot project on the development of a low-cost model for elementary computer education, Ms. Torgerson has taught Logo to third, fourth and fifth grade students as well as to teachers and parents.

Ms. Torgerson has also participated in the development of curriculum materials for Indian children. She has published "The One Who Sits on Top" and "Mean Mountain Sheep," two Indian creation myths as told by Johnny Arlee.

Other interests for Ms. Torgerson include skiing, hiking in the mountains and bicycling.

Mary Kay Owens Kriley entered the computer field as a programmer for Boeing Airplane Company. A magna cum laude graduate in mathematics from Seattle University, she has taught high school in Alaska and Seattle. She has also developed a computerized system, written in COBOL, for the Mountana Institute of the Arts.

Ms. Kriley has worked with middle school students in the computer classes at Rattlesnake School in Missoula, Montana. In addition to participating in her school district's computer curriculum committee, she has also taught at a computer camp sponsored by the University of Montana.

Her interests include skiing and sailing with her husband and three daughters.

Janet Teem Stone teaches seventh grade at Frenchtown Elementary School in Frenchtown, Montana. She uses a computer in her classroom as part of the art curriculum.

Ms. Stone became interested in Logo as a parent, and she participated in the pilot project at Paxson Elementary School.

Ms. Stone enjoys skiing and activities involving her husband and four children.

Graphics by Percy Franklin

LOGO IN THE CLASSROOM

Copyright © ICCE 1984

Table of Contents

Preface 4

Introduction 6
What Is Logo? 6
How Does Logo Fit into the Curriculum? 7
Why "Play Turtle?" 7
How Do I Get Started? 7
What Do I Need to Know About? 10

Scope and Sequence Chart 12

Session 1 13
To introduce students to the Logo primitives FORWARD, BACK, RIGHT and LEFT through self-involvement in spatial relationship activities.
To reinforce following and giving precise directions.

Session 2 17
To introduce abbreviated forms for the Logo primitives.
To provide practice in using the Logo primitives as spatial commands through self-involved activities.

Session 3 21
To introduce students to the computer and to the Logo Turtle.

Session 4 25
To examine how we change directions by turning, or pivoting.
To introduce the protractor as a tool for measuring the amount of turning.

Session 5 32
To introduce the concept of writing a set of directions.
To reinforce the concept of measuring the amount of turning.
To provide practice in using the protractor.

Session 6 35
To reinforce measuring "the amount of turning" in an angle.
To review metric measurement relative to Turtle steps.

Session 7 38
To introduce and give practice in using a small, desk-size protractor.
To introduce the concept of "angle" and relate it to the amount of turning in changing directions.

Session 8 45
To introduce the Logo primitives PENUP and PENDOWN, and the primitive for erasing.
To help students develop concepts of distance in Turtle steps and in angles by the amount of turning.
To provide practice for internalizing the Logo primitives.

Session 9 70
To introduce the concept of the teaching or editing mode.
To introduce students to writing procedures.

Session 10 78
To introduce the editing mode.
To give students practice in editing procedures.

Session 11 83
To introduce the word "debugging" to refer to finding and correcting errors in procedures.
To give students practice in using the editing features in Logo.

Session 12 90
To introduce the concept of one rotation as an amount of turning which is equal to 360°.
To reinforce the concept of procedures as a means of teaching Turtle to follow a set of instruction.

Session 13 97
To introduce saving and recalling of Logo procedures.
To help students develop an understanding of the computer's process of recording information on cassettes or disks.

Session 14 107
To introduce students to problem solving in Logo.
To introduce the primitive for repetition.
To reinforce working with angle measurement.

Session 15 110
To teach students to use procedures (subprocedures) within a procedure.

Session 16 120
To introduce the concept of recursion.

Session 17 128
To allow time on the computer for free exploration and creativity.
To allow time for assimilating and internalizing Logo primitives and geometric concepts at the child's individual developmental level.

Session 18 132
To introduce students to the use of variables as size input for geometric shapes.

Session 19 140
To introduce/reinforce creating circles of various sizes on the computer.

Session 20 147
To help students realize that they have made a tremendous amount of progress.
To make students feel good about themselves.

Charts 157
Keyboard Charts 158
Computer Systems Charts 164
Time-On Chart 169
Suggested Use of Time-On Chart 170
Partner Chart 171
Suggested Use of Partner Chart 172
Primitive Commands Chart 173
Suggested Use of Primitive Commands Chart 174
Large Protractor Master 175
Using the Large Protractor Chart 177
Color Charts 178
Key Functions for the Edit Modes 183
PUTTILE Chart 189
Graph Paper Master 190

References 191

Answer Appendix 192

Preface

During the summer of 1981, I attended San Diego State University's Computer Institute where Professor Seymour Papert introduced us to Logo, a computer language developed around a philosophy of learning. Papert, leader of the MIT research group which developed Logo, proposed that the computer could become a learning vehicle by allowing interaction with objects that are related to meaningful knowledge in the child's cultural environment. In Logo, a small, movable triangle/figure on the computer screen, called the turtle, becomes the vehicle for communication between child and object. Papert was explicit when he stated that the turtle is not a cure for the problems we find in education, but is a ". . . model for other objects, yet to be invented. My interest is in the process of invention of 'objects to think with,' objects in which there is an intersection of cultural presence, embedded knowledge, and the possibility for personal identification."* Papert's colleague, Swiss psychologist Jean Piaget, also emphasized that children learn best not from listening to lectures, but by acting upon objects in their environment.

My post-graduate studies in early childhood education were influenced by Piaget's theories of development, so I was intrigued by this new computer language that was presented as a means to develop intellectual structures.

Upon my return to Montana, I purchased Papert's book, *Mindstorms: Children, Computers and Powerful Ideas*, in which he describes his vision of children using computers. I became totally absorbed in thinking about this philosophy as I read and reread *Mindstorms*. But I came back to earth with a resounding THUD when I returned to the classroom that fall. Taxes, inflation and a dozen needs in every classroom precluded the purchase of even one computer for our K-5 school. Any computers purchased by the school district were destined to be used in the 6-8 grade computer centers.

Dr. Benjamin Bloom's well-known compilation of over 1,000 research studies has shown the importance of a child's younger years in the development of what we term intelligence. I believed Logo could strengthen this development. Many truly innovative educational steps in the elementary classroom are first tried cautiously in the grassroots tradition. I purchased a computer and brought Logo into my fourth-grade classroom, so I could see first-hand what Logo could do for children. With countless trials, errors and frustrations, interspersed with an ample dose of laughter, we explored with the turtle and soon became good friends.

By the end of the year, my students had put together a graphic display created by many short programs, called *procedures* in Logo. Each fourth-grade student wrote one of the procedures which were combined into a super-procedure entitled "Missoula." The program shows a busy Missoula street on the video display. Several cars drive by, a man walks down the street, a helicopter flies across the sky, and a dog lounges under a tree in the park. As the sun sets in the west, night comes to Missoula. Stars appear in the sky and the sun becomes the moon. Everyone goes home except one truck which continues to drive the lonely street throughout the night. As the sun rises in the east, the town again comes to life.

The students entered this project in a computer contest. We won second prize! This honor not only earned a computer for Paxson School, but served to strengthen the students' belief in their ability to work with computers.

At this point, I knew I wanted more training in Logo, so I attended the summer session at Columbia University to study the Logo language. During the rest of the summer, I sorted through the past year's notes, anecdotal records, activities, daily schedules and curriculum plans in preparation for a new school year. This planning had a special focus because our district had given approval for a pilot project using Logo in our school. Compiling the previous year's notes and plans proved to be an enormous task, so I enlisted the assistance of two colleagues whose backgrounds include teaching experience at the elementary and middle school levels as well as programming experience in the business world. Mary Kay Kriley and Janet Stone joined me as microcomputer buffs with an ever-increasing interest in Logo.

This packet is the outcome of that effort and the previous year's classroom experience. We believe that using Logo is more than learning a language and more than exploring with the turtle. The sessions outlined in this packet use the computer as a medium to explore other areas of learning. Hopefully, some of these activities will suggest other ideas and the use of Logo in other subject areas in your classroom.

We would like to hear how these sessions work in your classroom. What should be changed? What can be clarified? With your input, perhaps this packet can grow to include contributions from other

*Seymour Papert, *Mindstorms: Children, Computers and Powerful Ideas*, p. 11.

schools. Perhaps we can develop into a group of teachers helping teachers, with get-togethers at conferences and conventions. We could begin an exploration into Papert's learning theories and/or a sharing of ideas through *The Computing Teacher.*

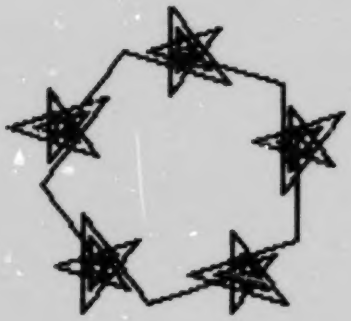
Papert and his colleagues at MIT have given us Logo to explore alternative methods of helping students learn. He stated that the turtle was only a model for the future development of objects which allow children to interact more with their environment, thus facilitating the development of their intellectual abilities. If this philosophy is to move into the classroom, it will be through the elementary school teacher.

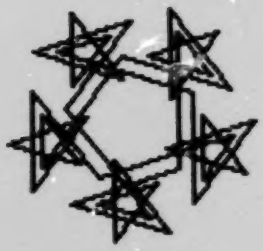
The materials in this packet are based on using Logo with the children at Paxson School in Missoula, Montana, and with students in the University of Montana's summer computer camp. Many volunteers and visitors offered ideas and encouragement as we floundered through the "getting-started" phase. To those who assisted in the classroom and at workshops, inservices and conferences, many thanks are extended. Appreciation is offered to our local School Board; Dr. Mike Vance, Curriculum Director; and to Michael T. Maxwell, Paxson School Principal. These people provided the opportunity to make the mini-pilot into a full Logo Pilot Project. Administrators who allow exploration with innovative methods and materials show true concern for the future education of children.

Logo is fairly new to all of us who are teachers of children, and at this time there is a lack of tried and tested Logo materials for classroom use. We were fortunate to find an experienced Logo teacher who could bring a fresh look to reviewing these materials. Tim Riordon of Alpine Educational Services in Eugene, Oregon, provided a valuable critique of the contents which led to further insights and clarification of a number of points. Rick Billstein of the University of Montana provided a valuable critique as a math educator and Logo enthusiast. Thanks to Gail Daggett we were able to include Atari Logo in the versions of Logo covered.

Throughout the months of preparing and testing this Logo material, the one other person who stands out significantly is Dick Ricketts. His foresight, viewpoints and encouragement brought this project into being. Without his assistance, this set of Logo materials would not be available for teachers interested in using Logo as a learning medium in the classroom.

—Shirley R. Torgerson





Introduction

A computer in the classroom sounds exciting!

- But what do I do with it?
- How do I fit one more thing into my crowded schedule?
- What should I drop out of the curriculum to teach "computers"?
- Aren't the "basics" important any longer?

This packet is a guide to help the elementary teacher make decisions about how to use a computer in a classroom setting. It is not a computer manual nor is it a tutorial on the Logo language. Manuals for the different versions of the language are supplied with the Logo packages. Excellent books written by the designers of Logo are also available.

This packet assumes that the teacher has some experience using computers in the classroom or has worked with Logo. However, these sessions can also be implemented by a teacher unfamiliar with Logo. In this case, the teacher should become familiar with the computer to be used, either by reading the instruction booklets supplied with the machine, or by getting help from a colleague who is familiar with the basic set-up and operation of the equipment. S/he should learn the proper procedure for powering up/down and loading Logo and become acquainted with the keyboard. The teacher who has little prior experience with Logo should work through the first few sessions before introducing students to the activities.

These ideas and techniques for learning Logo were developed to meet the needs of students in a classroom where computers were in short supply. The concepts and activities were interwoven with other areas of the curriculum. These activities were conceived and nurtured within the classroom, where learning was an ongoing process encompassing all subject areas. Many of the ideas are presented and learned most effectively in a regular classroom environment, rather than in a separate computer room.

In *Mindstorms*, Dr. Papert strongly advocates that schools provide students with their own microcomputers throughout the K-12 school years, because Logo is learned best at the computer in a highly individualized setting. Elementary teachers realize that this is currently unrealistic. Yet we want our students to receive rich and valuable computer experiences. These sessions have been developed for teachers who are faced with the dilemma of how to use a single computer effectively with a large group of students. Activities are group-oriented and structured only in the sense that students are guided through planned, computer-related experiences. We strongly recommend that the actual time individual students spend at the computer be as free as possible from teacher design. Exploration sparked by the student's own imaginative and creative energies is central to the Logo philosophy.

What Is Logo?

Logo is a computer language developed at Massachusetts Institute of Technology under the leadership of Dr. Seymour Papert. In its present form, Logo creates a user-oriented learning environment with turtle graphics, sprites and list processing.

Turtle Graphics

In the turtle mode, a small triangle/figure is in the center of the screen. This turtle moves about the screen making line drawings in response to simple program instructions. Since young students tend to identify with the movement of the turtle, the activities in this packet are first introduced as spatial movements in which the student acts out the role of turtle, following commands of FORWARD, BACK, RIGHT and LEFT. As the students later direct the turtle's movement about the screen, they are able to project themselves onto the triangle/figure, an idea Papert calls body knowledge, or body syntonicity. This idea is also fundamental in Piaget's work. His observations have shown that children perceive and create abstract ideas based on their previous experiences of direct, personal interaction in their environment. As the child develops this body knowledge by performing movements in space, insight is gained which enables the child to transfer this movement to the turtle's movement. Papert has called this "playing turtle." This idea is especially helpful when searching for the line in a Logo program (procedure), which makes the turtle perform an unexpected action. Such errors are referred to as "bugs." Debugging a procedure is greatly facilitated if children know how to play turtle; they are not simply *observing* the unexpected turtle movement—they are *experiencing* it.

Sprites (not available in all Logo versions)

The sprite mode enables students to create colorful, animated graphic displays by giving attributes of shape, color, heading and speed to one or more objects as they develop ideas of the interaction of time, distance, speed and velocity. Papert suggests that children have difficulty learning physics because the environment in which children develop doesn't provide them with enough concrete, hands-on, interactive experiences with such abstractions.

List Processing (not available in all Logo versions)

The list processing mode of Logo allows students to explore the manipulation of words and sentences. Students are able to create lists of information that

have meaning for their own age group. They can make lists of the books they have read; lists of their friends; lists of their math grades; lists of the teams their school plays (complete with scores). Through simple recursive commands they can add to or delete information from the lists. Information can be passed back and forth among lists, and arithmetic operations can be performed. By creating short procedures, students can manipulate information that is important to them. While working with list processing, they are developing skills to categorize and structure sentences and to gain insight into logical branching.

Logo is an extensible language. It begins with a few words called "primitives"—words the computer already knows. Students then use these primitives to expand the computer's vocabulary by adding new words through simple programs called "procedures" in Logo.

In a Logo environment, students are free to explore and interact with the computer using the turtle, sprites and list-processing capabilities. While interacting with the computer, their discoveries strengthen their developing knowledge bases. By integrating the knowledge students already have with new ideas from other curriculum areas, the classroom can become a Logo environment that facilitates a child's intellectual growth.

Logo in the Classroom introduces students to turtle graphics. Teachers will need to consult other sources for information on sprites and list processing.

How Does Logo Fit into the Curriculum?

The factual knowledge base in the world is increasing at such a rapid pace that our educational focus has moved away from memorizing facts to learning how to access and process information. Teachers are now emphasizing the development of intellectual structures to facilitate comprehension, problem solving and communication. At the same time, a strong academic foundation in basic skill areas precedes and accompanies the development of internal intelligence patterns. Logo, as used in this packet, strengthens basic skills while providing a medium for developing these important intellectual structures.

A scope and sequence chart shows how each session integrates subject matter areas. Basic skill areas in mathematics and language arts are an integral part of learning Logo through these materials. Learning the concepts of and developing skills in spatial relationships, patterning, estimating, sequencing, randomizing, probability, measuring and geometry, as well as developing problem-solving strategies and communication skills are emphasized. Turtle graphics presents a new way to approach geometry. Our experience has been that Logo provides a clarity of understanding for students that surpasses any other methods tried in our years of teaching at the elementary level.

Logo, as presented in this booklet, is not an isolated subject called "Logo" or "learning about computers." Logo is a language for learning subject matter, developing skill areas, solving problems, building intellectual structures, exploring language and communication—and it's just plain fun.

Why "Play Turtle"?

Many of our present teaching methods in elementary schools are based on the observations and research of Jean Piaget. Piaget's research stresses the necessity of children being involved in physical manipulation of objects to build their intellectual structures. Spatial relationships become meaningful to children because of their interaction with the environment. Children develop concepts of direction because of their own movement in space. They develop concepts of distance because of their own actions and involvement in movement. Self-involvement by children in spatial activities facilitates understanding movement of the turtle in Logo.

Turtle geometry begins with a small, triangular or turtle-shaped, cybernetic figure on a video screen. This figure is called the "turtle." During the development of Logo, the original turtle was a small robot that was electronically programmed to move about the room. The robot would draw lines as it moved, or pick up its "pen" and move without drawing. With the advent of low-cost, powerful microcomputers, the robot became the figure on the video screen. Using Logo, students enter commands for the turtle to move on the screen, drawing lines or not. (The robot turtle is also commercially available.)

Following the ideas of Piaget and Papert, it is our firm belief that the programming activities of children must be preceded by real spatial movement activities. They must identify strongly with the turtle as it turns, rotates and darts about the screen. Therefore, most of the teaching strategies in this packet address the child's ability to use self as a point of reference in spatial relationships. The younger the child, the greater need there will be for "playing turtle."

In later sessions, as more complicated procedures are designed for the turtle, a turn in the wrong direction can produce very different directional movements than were planned. When the "bug" (an error, in computer jargon) becomes difficult to find, teachers should recommend that students "play turtle." Errors can be discovered more easily when students walk out the commands.

How Do I Get Started?

Not all classrooms are the same, nor will all Logo situations be the same. The techniques in this sec-

tion are suggested models and should be adapted to the needs and interests of your class. Experiment! Try various management methods until you find those that best fit your daily schedule, your teaching strategies, and your learning objectives.

A Few Tips on Physical Arrangements

If you are able to keep at least one computer in your classroom, place the computer in a corner or nook of the classroom to offer privacy for those children who may be hesitant about sharing their work with others.

If the computer will be used while other classroom activities are carried out, then turn the monitor screen away from the class. Some children have a difficult time developing the self-discipline required to keep on task and are distracted by on-screen activities.

A shortage of computers may require sharing a computer, with one or more classes rotating daily, weekly or monthly. When the computer is out of your room, set up a computer work center in a corner where paper and pencil activities related to the computer can be carried out. Such a work center encourages continuity by keeping computer-related activities visible, even when the computer is not directly available. The computer corner should be equipped with a bulletin board and at least three chairs. Children often benefit from working together.

Selecting a computer corner involves checking the location of electrical outlets to be sure power cords will not cross walkways. A line filter, an electrical device that plugs into the 110 outlet on the wall, is highly recommended because it will control power surges that may destroy programs currently being used in the computer. The line filter usually filters electricity from the wall outlet through a small box; it provides two or more outlets for plugging in the computer and components. If a line filter is used, only one wall outlet per computer should be needed. If the computer keyboard is not attached to the screen, allow about 30-50 centimeters between the keyboard and the screen. If students are using the computer in small groups, this will help them see what is happening on the screen and minimize eye strain. (Research is inconclusive about the health effects of working on cathode ray terminals, which do emit low-level radiation similar to ultraviolet lighting. Most students will be using the terminals for short periods of time.)

Students should be instructed in the proper care of the computer and its accessories. Much of the time the computer corner will not be strictly supervised by an adult, so students will be responsible for the proper care and use of the machine. Class discussion should lead to a set of guidelines for computer use. Students should know how to set up the computer in the morning and how to turn it off at the end of the day. Food and drink should not be allowed near the computer—no exceptions! Finally, if your computer is in a carpeted area where static electricity is a problem, you may want to purchase anti-static spray for the carpet. Use your computer manual for more detailed advice about the care of your computer and disks.

The Daily Schedule

This model works well—however, be sure to build in flexibility as you adapt it to your class situation.

A Logo session can be presented once a week to the entire class and need take only about half an hour. This includes a class activity and a follow-up problem or practice assignment that can be completed during the session or at some other time during the week. Considering the integrated subject matter that is being taught/learned with new Logo skills, time taken from the "basics" is minimal.

After presentation of the weekly session, students can schedule time on the computer by using the Time-On chart (see Charts section). The amount of free time each student has on the computer will depend on your daily schedule and the number of computers available. A Time-On chart should be made for each computer.

1. Mark the Time-On chart into half-hour slots that encompass the beginning and end of your school day. These slots can be extended or changed at any time if you laminate the chart and use water-base markers or crayons. The chart can also be photocopied and posted on the wall each week.
2. Cross off any time slots which will not be available to the students that week; for example, times reserved for lunch, physical education, music, library, or the introduction of new concepts in other areas which require full student participation, etc.
3. Ask each student to sign his/her name on one open slot. Devise a system that ensures that the same student does not always get first choice. After each student has signed up for one open slot, allow a second turn if there are enough open slots for all. If not, the open slots can be given out as doubles—one slot for a pair of students. If possible, keep the computer in use all day.

During the student's Time-On, s/he must first do any required work from the weekly session. The required work is meant to reinforce the concepts that were introduced. After completing the required work, the student is free to explore as desired. Students must be cautioned not to disturb the rest of the class or interrupt instruction. Since help from the teacher is often unavailable, students soon realize that exploration of different avenues is necessary to solve some problems. Students are responsible for moving to and from the computer without

disrupting other activities.

One of the questions teachers frequently ask is, "Won't students working at the computer miss important assignments and explanations?" This problem can be minimized by using the Partner Chart (see Charts section). Post the Partner Chart and allow students to select a partner, or pair students according to a sociogram and teacher judgment. A partner's responsibilities include collecting any worksheets passed out while the other partner is on the computer, writing down assignments, and turning in any papers due at that time. This system becomes a habit with students and works as well for other absences by the partner. The partner is responsible for explaining assignments, but not for teaching concepts. For more details, see "Suggested Uses" of the Partner Chart in the Charts section.

Volunteers

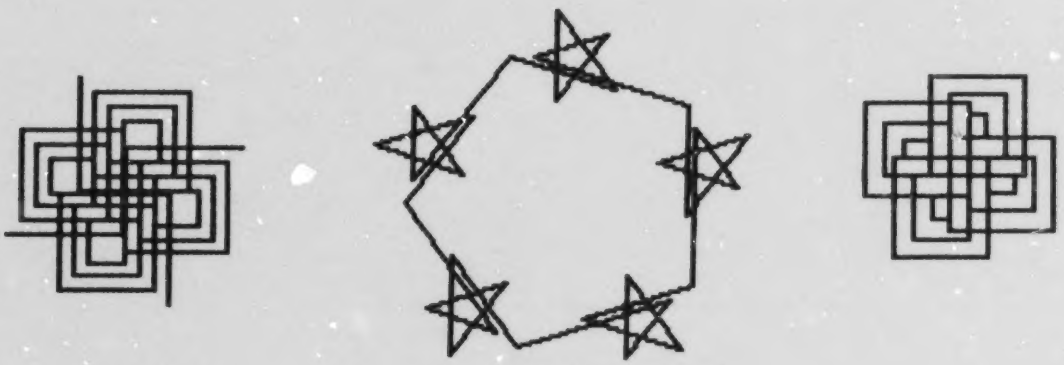
If you decide to use volunteers to assist students working on the computer, they should fully understand that their usual role is helping students by asking questions that lead them to discover their own answers. Without training, volunteers may not realize that "showing how" at the keyboard transforms the student into a passive observer instead of an active learner. The volunteers should always let students do the typing. Training may take place in a discussion period prior to volunteer activities or through a written set of guidelines. Often, students merely need someone to reflect their own thinking in order to gain a new perspective. Sometimes a hint or suggestion may guide a frustrated mind down a new path, or the student may need reassurance that it's all right to try something unusual. How reticent volunteers should be depends on the particular group of students and their learning needs. A good guideline is, "When in doubt, don't intervene."

Our experience has shown that excellent assistance also comes from other classmates. Since grading is not a part of the Logo experience, competition seems not to enter into the learning—students simply enjoy sharing and helping each other. Encourage students to ask each other for possible solutions to problems.

Record-keeping

Students may want to keep copies of their procedures in either a spiral or loose-leaf notebook. Loose-leaf notebooks are more adaptable and can be cleaned out easily. Individual file folders are more practical for younger children who may have trouble keeping track of notebooks. The file folders can be placed in a cardboard box that has been covered with contact or construction paper. The box should be located in the computer corner. This provides ready access for the teacher and students.

Logo primitives ("words" the computer knows) are presented to students so that they can explore the learning environment Logo provides. Testing or grading standards for Logo are not included. Completion of the individual worksheets can be used to assess the mastery of the concepts and skills presented.



What Do I Need to Know About Using These Materials?

Logo is a language that was created by a research group at MIT. As Logo has become more and more popular, a number of companies have created their own versions of the language. While all of these versions of Logo contain the same general ideas about turtle graphics, they differ considerably in detail.

This packet has been designed to be used with any one of five distinct versions of Logo. The chart below has five columns, one for each of the five versions of Logo illustrated in this book. (If your Logo version is not in the chart, you can still use these materials. However, you will need to refer to your Logo reference manual for details on each primitive as it is introduced.)

Notice that the primitives FORWARD, BACK, RIGHT and LEFT are the same in all five versions of Logo covered in this packet. Their use is discussed beginning in Session 1. Later a number of additional primitives will be introduced. The words for these primitive as well as details on how they work will differ among various versions of Logo. Note that inputs needed for each primitive are not listed. You should refer to your Logo manual for details covering use of each primitive.

PRIMITIVE CHART:

	APPLE	ATARI	MIT	TI	TRS
Logo Version	Logo Computer Systems, Inc. (L.C.S.I.)	Logo Computer Systems, Inc. (L.C.S.I.)	Terrapin & Krell	Texas Instruments	Tandy Radio Shack
Computer	Apple II (64K) Apple IIe	Atari 400 Atari 800	Apple II (64K) Apple IIe Commodore 64	TI 99/4A	TRS-80 Color Computer TRS-80 Color Computer 2
Primitives	FORWARD BACK RIGHT LEFT	FORWARD BACK RIGHT LEFT	FORWARD BACK RIGHT LEFT	FORWARD BACK RIGHT LEFT	FORWARD BACK RIGHT LEFT

Find your computer and Logo version on the chart. The word or abbreviation at the top of that column identifies that version of Logo in each session. Those sessions that introduce new primitives list them for each version. You may want to highlight your version and primitive using a read-through color pen. Separate student pages are listed for each version whenever there are differences, and separate teacher pages are sometimes used to avoid confusion. Most sessions are readily adaptable to any of the five versions. However, some of the activities are not possible in all versions of Logo.

Objectives, Concepts/Skills, Facilities/Materials and Preparation sections begin each session. Any new Logo primitives are listed along with comments on the session. The Preliminary Activity section usually reviews previous skills/concepts or Logo primitives. Sharing and discussing discoveries or problems is always encouraged during the preliminary activity. The Activity section outlines the lesson in detail, and is followed by the Assignment section, and then by any worksheets required for the assignment as well as any masters or procedure listings needed for the activity. Worksheets include Problem Sheets, Practice Sheets and Challenger Sheets. The Problem and Practice Sheets can be assigned immediately after the activity, or as students have time after completing other assignments. You may want to discuss or check some of these worksheets as students progress. Individual teaching styles will determine when and how these worksheets should be followed up. The Challenger Sheets are for those students who finish the basic assignments and are ready for more complex work. An Answer Appendix for most of the worksheets is included in these materials.

Logo primitives are presented to students during the activity. Some students will not immediately use all of the primitives which have been introduced. Adding new commands to an ever-growing list and keeping these posted near the computer will give students access to the language when they need it. A Primitive Commands chart is included in the Charts section.

This packet is designed as a resource guide. Sessions do have a sequential development, but they are not meant to be used as a grade-level textbook. The activities are designed as an introduction to the Logo language, based upon other subject matter appropriate to general grade-level guidelines. The previous depth of understanding in these related areas depends upon the background of your students.

Concepts of each session can be used to introduce material or to reinforce previous learning. Many concepts may not be new to some students, but previous learning can be built upon. These resource materials should be adapted to the wide variations in student abilities as well as the developmental needs and temperament of your class.

The concepts and activities included are perhaps most useful for grades three through six. As with any other resource book, select, alter and use the ideas that will give your particular class the experiences needed to become excited, self-motivated Logo users.

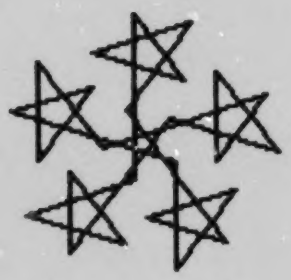
Some of the activities may have to be modified because of the physical setting or other limitations. Adaptability and creativity may be needed. If the weather cancels playground activities, try the gym or the hallway. Try tomorrow! But keep students involved in actual physical experiences.

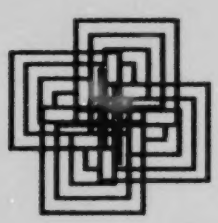
The primary and most important goal of these materials is to provide students with a successful and stimulating computer-oriented learning experience. Once a few Logo primitives become easy to manipulate, students working at the computer can become involved in the creative and heuristic learning environment discussed by Seymour Papert. In essence, the child and the computer will create a context in which significant intellectual discoveries can occur. As students learn, they must have the opportunity to make their own decisions regarding their explorations at the computer. Students will develop confidence as they set their own problems and goals and as they devise methods and procedures for attempting to solve those problems.

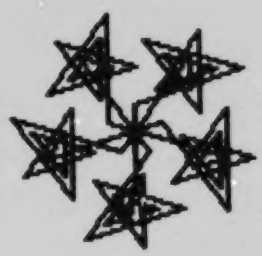
During the development of this packet and the use of these materials with elementary students, the Logo language was presented regularly and sequentially to all students. The students enjoyed the activities and completed assignments in ways that reinforced Logo concepts. However, during their Time-On at the computer, students were self-directed, and individual variations in interests, personalities and developmental levels soon became apparent.

One little girl, very much a happy, easy-going child, completed the assignments in progressive order with the class. In her free time, she always returned to the recursive poly procedures, delighting in the fantasy of changing shapes and colors and enjoying the abstract beauty of her designs. Her interests contrast sharply with another, more competitive student. This student set high goals and pushed herself to accomplish her tasks. Whatever the task given, she found a way to make a simple task more complex and challenging. Needless to say, at the same time her skills improved, she suffered more frustration and sought help from the teacher more often than many other students.

Finally, we offer a caution against giving children more of the langauge than they are developmentally ready to handle. If teachers let students set their goals too high or give them too much of the Logo language too soon, the resulting learning experience is more likely to be frustrating for most students, and the teacher's role is more likely to be reinforced as a problem solver rather than as a facilitator.







SCOPE AND SEQUENCE CHART
INTEGRATING LOGO IN THE CLASSROOM

EDUCATIONAL GOALS	SESSION:																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Strengthening creativity and discovery-learning abilities.			●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Developing problem-solving skills.		●								●	●	●	●	●	●	●		●	●	●
Strengthening concepts of spatial relationships.	●	●	●	●	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●
Developing procedural thinking abilities (logic, sequence, order).	●	●		●	●	●	●		●	●	●	●		●	●	●	●	●	●	●
Developing intuitive reasoning and estimating skills.						●		●				●		●	●	●		●	●	●
Developing computer programming concepts.			●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Strengthening mathematical concepts and arithmetic skills.	●	●		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Strengthening geometry concepts.	●	●		●	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●
Strengthening measuring skills, linear and angular.	●	●		●	●	●	●	●	●	●	●	●			●	●	●	●	●	●
Strengthening language arts concepts.	●	●			●	●			●	●	●	●	●	●	●	●	●	●	●	●

Session 1

*"The child cannot . . . reconstruct a perspective point of view until he has himself occupied the position to which it corresponds. This is because thought can only replace action on the basis of the data which action itself provides."**

Jean Piaget and Barbel Inhelder
The Child's Conception of Space

OBJECTIVES: To introduce students to the Logo primitives **FORWARD, BACK, RIGHT** and **LEFT** through self-involvement in spatial relationship activities and to reinforce following and giving precise directions.

CONCEPTS/SKILLS: An understanding of commands as directed movement in space using self as a point of reference and conceptualization of commands for future identification with the turtle.

FACILITIES/MATERIALS: Overhead, chalkboard or tagboard strips labeled FORWARD, BACK, RIGHT, LEFT; playground (gym, hallway, etc.); 5x8 card, envelope and sheet of paper for each group of 4-5 students; student copies of Problem Sheet.

PREPARATION: On 5x8 cards, write directions for finding envelopes in various places on the playground, (gym, hallway, etc.). Number corresponding cards and envelopes. For each envelope, prepare a sheet of paper with an assignment to write directions to another specified location. (Sample 5x8 card and Assignment Sheet follow the Activity Section.)

PRIMITIVE CHART:

	APPLE	ATARI	MIT	TI	TRS
Logo Version	Logo Computer Systems, Inc. (L.C.S.I.)	Logo Computer Systems, Inc. (L.C.S.I.)	Terrapin & Krell	Texas Instruments	Tandy Radio Shack
Computer	Apple II (64K) Apple IIe	Atari 400 Atari 800	Apple II (64K) Apple IIe Commodore 64	TI 99/4A	TRS-80 Color Computer TRS-80 Color Computer 2
Primitives	FORWARD BACK RIGHT LEFT	FORWARD BACK RIGHT LEFT	FORWARD BACK RIGHT LEFT	FORWARD BACK RIGHT LEFT	FORWARD BACK RIGHT LEFT

COMMENTS:

One of the central ideas in using Logo to explore geometry is that the turtle uses itself as a point of reference as opposed to using a Cartesian coordinate system. Students' actual experience in walking the commands will help them later in identifying with turtle movements on the screen, especially as the graphic displays become complex. The younger the student, the more essential the self-involvement in spatial activities. Some variations of this activity are listed at the end of this session—however, since debugging a procedure is often facilitated by knowing how to "play turtle," some walking of the commands is recommended.

At this point students have not been introduced to the turtle, so they do not know that they are learning to "play turtle." What they are learning is that four commands can be used to move themselves anywhere on the ground. In addition they are practicing giving and following precise directions in preparation for writing and debugging procedures.

In helping your students learn Logo, there are two important instructional ideas to keep in mind. First, new ideas need to be related to and closely tied in with ideas the students already know. Second, the

*Reprinted by permission of Humanities Press Inc., Atlantic Highlands, NJ 07716.

number of new ideas to be presented in one session must be quite limited.

This first session illustrates these two ideas. The words FORWARD, BACK, RIGHT and LEFT are familiar to students. The session consolidates this knowledge, getting students used to the idea of following and giving directions made up from these words.

In Logo, FORWARD and BACK are always accompanied by a number telling how many turtle steps the turtle is to move forward or back. Thus, FORWARD 25 and BACK 18 are correct commands in Logo. Students have knowledge of linear measure and can readily deal with commands such as "take three steps forward," or "take two steps back." This makes it convenient to introduce FORWARD and BACK in a manner both easy for students to understand and consistent with how Logo functions.

The commands RIGHT and LEFT in Logo must be accompanied by a number telling how many degrees to turn. Thus, RIGHT 45 means turn 45° to the right, while LEFT 90 means turn 90° to the left. Many students are not familiar with angular measurement. Thus, this first session introduces the words RIGHT and LEFT but without a specification of how many degrees. The word RIGHT is to mean a full 90° (right angle) right turn while LEFT is to mean a full 90° left turn. The difficulties of angular measurement are left to a later session.

The commands RIGHT and LEFT only change direction in place; there is no movement forward or backward. It may be necessary to call attention to this. Students usually turn 90° when asked to turn RIGHT or LEFT. Make sure they use a 90° turn when following RIGHT or LEFT commands, but do not call attention to it unnecessarily at this time.

PRELIMINARY ACTIVITY:

Print the words FORWARD, BACK, RIGHT, LEFT on the overhead or chalkboard, or use labeled tagboard strips. Discuss the meanings with students.

ACTIVITY:

Part 1

Ask the students to stand, move FORWARD 2 steps, BACK 1 step, turn RIGHT (emphasize that there is no forward or backward movement), FORWARD 1 step, BACK 2 steps, turn LEFT, then FORWARD 2 steps, etc.

Explain to students that they can move from one place to any other by using only these four commands. Ask one student to stand. Give commands to direct that student about the room to various locations. Then let a student give the commands for moving another student to a specific location.

Students also need to see the written commands. Write on chalkboard, calling attention to the single line, vertical listing format; use of capital letters; and the END command:

```
FORWARD 6
RIGHT
FORWARD 3
RIGHT
BACK 1
RIGHT
FORWARD 3
LEFT
FORWARD 6
END
```

Have a student walk the directions. When students demonstrate an understanding of how the directions are written and followed, go on to Part 2.

Part 2

Have each student bring a pencil, and then form them into groups of four or five. Number the groups. Tell them that each group will be given a 5x8 card with directions to a different location on the playground (or gym, hallway, etc.). If they follow the directions correctly, they will find there an envelope with their group number on it. Inside the envelope they will find an assignment asking them to write a set of directions to another specified location.

Pass out each group's 5x8 card. Make students aware that the directions on the card are given using only FORWARD, BACK, RIGHT, LEFT.

Since each person's steps are different in length, tell the students just to walk normally, and they will come fairly close to their envelope. When they find it, they should open it, read it and complete the assignment. Suggest that when they write the commands FORWARD, BACK, LEFT or RIGHT, they should use a new line for each command, similar to the directions given on the 5x8 card.

Variations for Part 2:

- Use a hundreds chart and a marker that can point in a certain direction. The marker might be a small picture of a turtle, with FORWARD being the direction the turtle's head is pointing.

  Place the marker on a square, facing the top of the board. Students are to write sets of directions to move the marker to other places on the board and/or through sequences of places on the board. For example, write directions to move the marker to numbers that add up to 387; move the marker to every number containing a 5; move the marker

to every number divisible by 3. Students should practice following the directions written by other students.

- Secure a supply of city maps. Have students begin at a specified point and follow directions of FORWARD 3 BLOCKS, RIGHT, FORWARD 6 BLOCKS, LEFT, etc. Students then identify the intersection by street names. They, in turn, can write similar directions for returning to the original position. (This activity would correlate well with a social studies and map skills unit.)

SAMPLE 5x8 CARD:

Group 1:
Start at tetherball pole facing the school.

FORWARD 20
LEFT
FORWARD 15
RIGHT
BACK 8
RIGHT
FORWARD 20
LEFT
FORWARD 8
END

Remember, each 5x8 card should include the group number and directions to a different location in the same general area.

SAMPLE ASSIGNMENT SHEET:

Complete the directions below using only the four commands FORWARD, BACK, RIGHT and LEFT.

Start at the point where you found your group's envelope, and go to the bike rack.

Begin by facing north. Tell how many steps you take when moving FORWARD or BACK. Remember to write only one command on each line.

Write END when your directions are complete. Sign your names at the bottom of this sheet.

DIRECTIONS TO BIKE RACK
FORWARD_____

_____
_____
_____
_____
_____

ASSIGNMENT: Have students complete the Problem Sheet for this session.

Session 1
Problem Sheet

Name _____
Date _____

"Will Hungry Mouse Reach the Food?"

Below is a 9 cm x 12 cm room. Just inside the door and immediately to the right is a wall extending 2 cm into the room. Behind this wall is a dish of food shown by the circled X. You are a HUNGRY MOUSE. You enter the room at the door. You can smell the food but have a problem. **You can only move FORWARD and can only turn LEFT.**

Draw your path to the food. Use a ruler to make straight lines and the end of your ruler to make LEFT turns.

If you succeed in reaching the food, try sending a message to your friend (?) HUNGRY CAT, who also has the problem of only going FORWARD and making LEFT turns. If your directions are correct, HUNGRY CAT can finish the food. If your directions are *not* correct, HUNGRY CAT will not find the food—but might find *you!*

Dear Hungry Cat,

FORWARD _____ cm
LEFT
_____
_____
_____
_____
_____
_____
_____
_____
_____

END _____

Your friend,
Well-Fed Mouse

20

© ICCE 1984. Reproduction for classroom use permitted.

Session 2

"Piaget asks the fundamental question—How does one abstract (develop a mental image of) a geometric shape? Is perception (looking at a shape) sufficient? These experiments indicate that perception or seeing is not enough. There must be a physical action of the child on the object . . . .

*"Piaget concludes: '. . . the children are able to recognize, and especially to represent, only those shapes which they can actually reconstruct through their own actions. Hence, the "abstraction" of shape is achieved on the basis of co-ordination of the child's actions and not, or at least not entirely, from the object direct (that is, by just looking at it).'"**

Richard W. Copeland
How Children Learn Mathematics: Teaching Implications of Piaget's Research

OBJECTIVES: To introduce abbreviated forms for the Logo primitives and to provide practice in using the Logo primitives as spatial commands through self-involved activities.

CONCEPTS/SKILLS: Internalization of primitive concepts using self as a reference point.

FACILITIES/MATERIALS: Overhead or chalkboard; student copies of Problem Sheets (2); graph paper for Challenger section (optional).

PREPARATION: Master for graph paper is in the Charts section.

PRIMITIVE CHART:

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT

COMMENTS:

Learning and forgetting are closely related. Some students learn more rapidly than others, and some students forget more rapidly than others. Each session should begin with a review of relevant ideas and a chance for students to ask questions.

PRELIMINARY ACTIVITY:

Review the activities from Session 1. On the chalkboard print the primitives FORWARD, BACK, RIGHT, LEFT. Give students time to discuss any problems or special incidents relating to the activities.

ACTIVITY:

Part 1

Introduce the abbreviated forms FD, BK, RT and LT, explaining that the shortened forms are easier and quicker to write. Relate to abbreviations in common use:

Jonathan	Jon
Elizabeth	Liz or Beth
Alexander	Al or Alex
United States	U.S.
Administration Building	Ad. Bldg.
Television	TV
Ante Meridiem	A.M.

Have students suggest other abbreviations.

*Reprinted with permission of Macmillan Publishing Company from *How Children Learn Mathematics: Teaching Implications of Piaget's Research*, Third Edition. Copyright © 1979 by Richard W. Copeland.

Part 2

Write directions using the abbreviated primitives to a point of interest in the school's vicinity. Use the school's front door as a starting point, and indicate the direction to head. Refer to this point as home base or home—the starting place of the mental journey in this session. Have students try to visualize the destination by mentally following the directions. For example, write on the overhead or chalkboard:

BEGIN OUTSIDE WITH BACK TO SCHOOL
FRONT DOOR
FD to sidewalk
LT
FD to corner
LT
FD 2 blocks
RT
FD across street
END

Where are you?

BEGIN OUTSIDE WITH BACK TO SCHOOL
FRONT DOOR
FD to sidewalk
RT
FD to corner
RT
FD 1 block
LT
FD 1½ blocks
END

Where are you?

Repeat with other directions if needed.

You may individualize the activity by giving closer end-points for students who have difficulty visualizing the destination. Use the inside of the school. Write directions from one room to the library or to the lunch room, so that students can "walk it"—that is, actually perform the action.

ASSIGNMENT:

Have the class write a set of directions using only the primitive abbreviations. Students are to use the school's entrance or some other place in the school neighborhood as a starting point and write directions to their homes. Remind them to indicate the direction they are heading and to write END as the last direction. The assignments could be posted on a bulletin board with the names covered and used as an extension activity in which other students try to guess where the directions lead.

Have students complete the two Problem Sheets.

When using the Problem Sheets, encourage creativity and divergent thinking skills. Some students may begin their directions by orienting the turtle in a reverse position before beginning, so the first movement is BK. All directions are acceptable as long as the objective is accomplished. Debugging through playing turtle is not formally introduced until a later session, but if the concept is useful here, introduce it.

CHALLENGER:

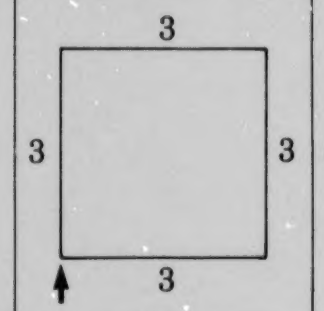
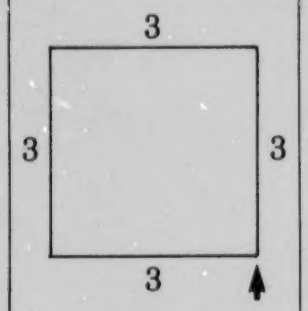
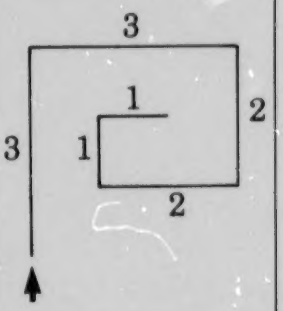
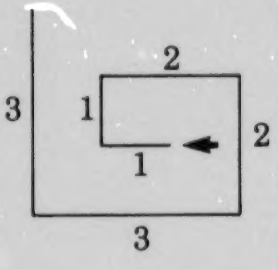
You may want students to do this activity after they complete their Problem Sheets:

Use a piece of graph paper to draw a helicopter view of a four or five-block neighborhood with stores, churches, homes, the school, gas stations, etc. On the map, using lines and arrows, draw three different ways to get to one building from a home or store. Then write the directions using the primitives FD, BK, RT, LT. Be sure to include END.

Session 2
Problem Sheet I

Name _____
Date _____

Can you write directions for telling someone how to draw these figures? Use the Logo primitives FD, BK, RT, LT. There may be more than one way to draw a figure! Begin your directions at the symbol "▲" and use END to show you are finished.

Figure 1	Figure 2	Figure 3	Figure 4
			
Directions	Directions	Directions	Directions

Draw another figure, using your ruler to make straight lines. The end of your ruler is a good guide for making corners. Use the back of this page to draw your figure and write directions for drawing it below that figure. Remember to put the symbol "▲" where you begin your directions.

© ICCE 1984. Reproduction for classroom use permitted.

Session 2
Problem Sheet II

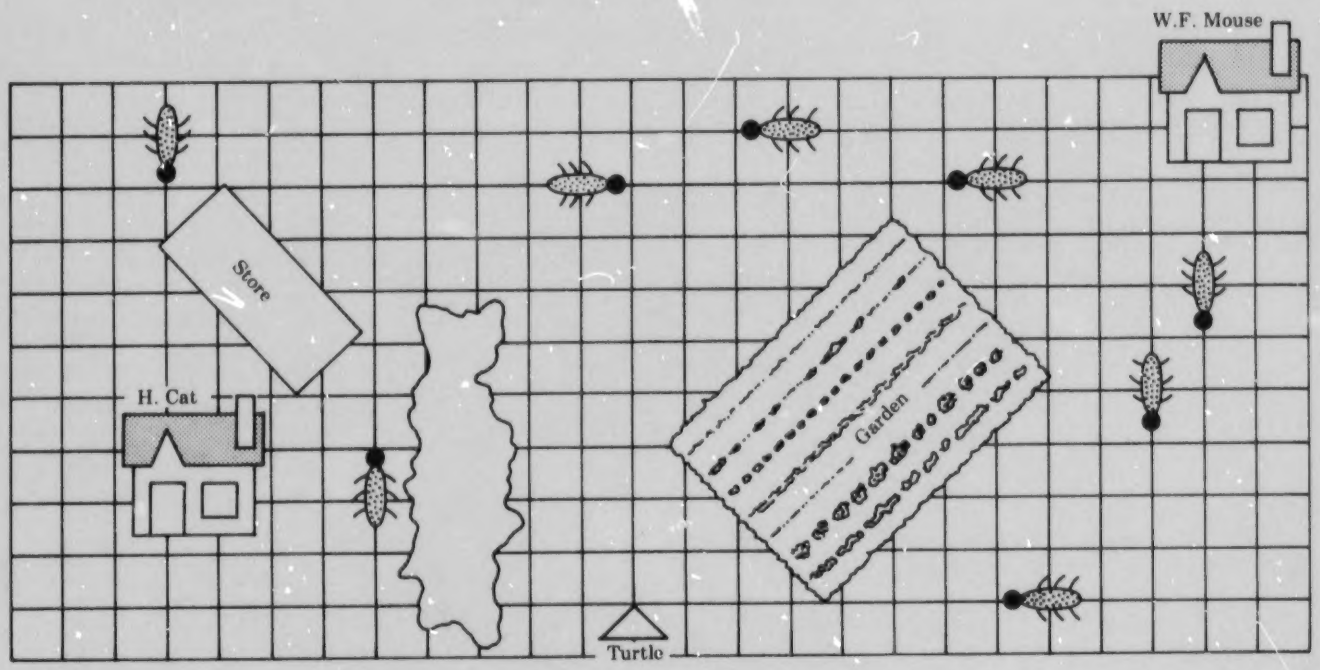
Name _____
Date _____

"Help Turtle Visit Hungry Cat and Well-Fed Mouse"

Below is a grid showing Hungry Cat's house, Well-Fed Mouse's house and a new friend, Turtle (the triangle). Turtle wants to visit both H. Cat and W. F. Mouse. You are to write directions for Turtle to follow to visit H. Cat. Then write a set of directions taking Turtle from H. Cat's house to visit W. F. Mouse. Turtle must not walk in the garden, in the lake or through the store. The bugs eat turtles, so don't touch them! Use the commands FORWARD, BACK, RIGHT, LEFT or the abbreviations FD, BK, RT, LT. Use the lines as streets.

EXAMPLE: TO HUNGRY CAT'S HOUSE
FD 2
LT
FD 3
END

Where would Turtle be?



© ICCE 1984. Reproduction for classroom use permitted.

Session 3

"The most novel and exciting idea for using computers to teach children turns the whole teaching concept around. The children teach the computer. Instead of the computer taking charge of each session by asking all the questions and allowing only simple answers, the student remains firmly in command. He uses the computer with the same kind of authority with which he uses a blackboard or a pencil. In fact, one interesting way in which children use the computer is simply as a very sophisticated pencil. The children learn how to tell the computer to make drawings for them: geometric figures, intricate designs and even rather complicated pictures.

*"At MIT, a mechanical 'turtle' has been designed that can easily be connected to almost any computer. A very simple programming language called Logo has been developed specifically for operating the turtle. Even 8-year-olds find it simple and fun to compose Logo programs."**

Bertram Raphael
The Thinking Computer: Mind Inside Matter

OBJECTIVE: To introduce students to the computer and to the Logo Turtle.

CONCEPTS/SKILLS: Awareness of computer components and functions and use of erase keys.

FACILITIES/MATERIALS: Computer with Logo module, disk or cartridge; student copies of the Keyboard sheet; Keyboard wall chart; Computer System wall chart; Primitive Commands wall chart; Time-On Chart (see Charts section); and student copies of the Problem Sheet.

PRIMITIVE CHART:

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT

COMMENTS:

Session 3 is the first "hands-on" session. Some students may have used a computer before. For others, this will be their first computer experience. It is important that this initial experience be successful and rewarding.

In introducing the initial computer experience, you will use some computer vocabulary such as monitor (or display screen or TV set), keyboard and memory. These terms will be used repeatedly in the future, so they will eventually become part of the students' working vocabulary. There is no need to place great emphasis on this vocabulary at first exposure.

The details for turning on a computer and loading Logo into primary memory vary with the particular computer system being used. Eventually each student will master these details, probably from other students. But for Session 3, you will want to do this for the class.

When a student comes to the machine, there may be a display left by a previous user. Thus, the first thing a student needs to know is how to clear the screen and prepare the machine to receive a new sequence of commands. Check the first row of the PRIMITIVE CHART for the command for your version of Logo and make sure each student learns that command. This is called entering the **draw mode.**

In Session 3 and a number of subsequent sessions, students will use Logo in an **immediate execution mode** (also called **immediate mode** or **command mode**). This means that each command they type is

*Bertram Raphael, *The Thinking Computer: Mind Inside Matter*, San Francisco: W. H. Freeman and Company, 1976.

executed immediately after they signal the end of a command by pressing the RETURN or the ENTER key. Of course, once execution begins the student cannot correct an error made in typing the command. If a student is drawing a detailed picture using a sequence of commands, s/he may be frustrated by a small typing error.

In this session students will learn to correct a typing error that is discovered before the machine begins executing a command (before the RETURN or ENTER key is pressed). In later sessions students will learn to write procedures—sequences of commands that are stored for later execution. They will learn to correct errors in these procedures. This will provide them with a powerful tool for drawing complex figures and for correcting errors made while working on complex problems.

This session is also a student's first introduction to specifying an angle in the RIGHT and LEFT commands. Session 4 contains work on measuring and using angles.

ACTIVITIES:

Note: Computer hardware varies greatly. Any information presented here will need to be modified to fit the computer and components you are using. The following information is presented as a rough guide.

Place the computer and components on a table in a convenient location for class demonstration. Load Logo into the computer. Have students sit in a semicircle with the back row standing to allow closer proximity to the computer.

Tell students that all the parts together are called a microcomputer system. Name each part as you briefly discuss its function. Also, call attention to the Computer System wall chart as each part is discussed. Tell students to call each part by its proper name.

1. **Central Processing Unit (CPU) and Memory:** This is the "brains" of the computer, where all the work is done and remembered if necessary.

2. **Monitor** (video display or TV set): The words and pictures appearing here are typed on the keyboard or loaded from disk, cassettes or cartridge. The answers or results from the CPU may also be displayed on the **monitor** screen. Compare the process to using paper and pencil to work a math problem. The calculations are not done **on** the paper; they are a reflection of the process in your head. The "work" is done in the brain and displayed visually on the paper. With a computer, calculations are done in the CPU (brain of the computer) and visually displayed on the **monitor**.

3. **Keyboard:** Pass out student copies of the keyboard sheet. Discuss letters, numbers, space bar and shift keys. Briefly discuss other keys that are special to your computer. Tell students that the important key for today is the key to erase a typing mistake before the computer carries out the incorrect command. Demonstrate erasing typing mistakes with the appropriate key for your version of Logo and computer. (See chart below.)

4. **Logo Disk or Module:** Contains a special language called Logo for communicating with this computer.

COMPUTER	LOGO VERSION	KEY FOR ERASING
Apple	L.C.S.I.	← Key (left arrow)
Atari	L.C.S.I.	"DELETE BACK S" Key
Commodore 64	MIT	DEL Key
Apple	MIT	ESC Key
TI 99/4A	Texas Instruments	FCTN and "3" Key (Hold down at the same time. The "3" key may be labeled ERASE.)
TRS-80 Color Computer	Tandy Radio Shack	← Key (left arrow)

Tell students that a few years ago some computer scientists made a little robot. By using special words, they taught the robot to move around the room. It could go FORWARD and BACK, and knew that if it touched something, it should make a turn. This robot had a pen and could draw lines as it walked about the floor. These computer scientists called this robot a "turtle," maybe because it moved slowly.

Later, a turtle robot was developed which had a triangle shape and knew many of the same words as the original robot. This triangle-shaped turtle moves about on a computer screen and knows the words FORWARD, BACK, RIGHT and LEFT.*

To talk to this turtle you must get the turtle's attention by typing:

APPLE: CLEARSCREEN (CS)
ATARI: CS (The abbreviation CS is the primitive in Atari Logo)
MIT: DRAW
TI: TELL TURTLE
TRS: R

Discuss the use of the RETURN or ENTER key to send commands into the computer's memory. Even though the words are on the screen, the turtle cannot carry out the commands until this key has been pressed. Demonstrate with several commands.

The turtle can turn, but needs to know how much to turn. Discuss with students how they have been using RIGHT and LEFT to mean a turn that looks like the corner of a square. Show them that RT 90 or LT 90 is that particular turn, and show them how the turtle can turn different amounts as students type in different numbers. Do not call attention to the 360° in a circle unless students offer this information.

Demonstrate several turns, emphasizing that a command for turning does not move the turtle FORWARD or BACK. The commands RT and LT are only meant to turn the turtle in a different direction.

There **must** be a space between the command and the number (FD 6 rather than FD6). If the space is omitted between the primitive and the number, the computer handles it as one command and will give the error message that it does not recognize that command.

Tell students that the same command for entering the **draw mode** can be used to clear the screen and start a new picture with Turtle in the center. (TRS users: Press ⟨BREAK⟩ before pressing R.)

Remind students that typing mistakes can be erased before a line is entered into the computer's memory. Demonstrate again using the appropriate erase key (listed under **3. Keyboard** above).

After the demonstration and question period, show the Primitive Commands wall chart. This chart should be displayed in the computer area for easy reference. Add primitives to the chart only as they are discussed. Conclude the demonstration period by putting up the chart.

ASSIGNMENT:

After the demonstration, sign up students for 15-minute blocks of time to explore with the turtle. You may want to introduce the Time-On Chart at this time. Directions and suggestions for using this chart are included in the Charts section. Two students may share their time for a total of 30 minutes, providing they both have an equal chance to use the keyboard to move the turtle.

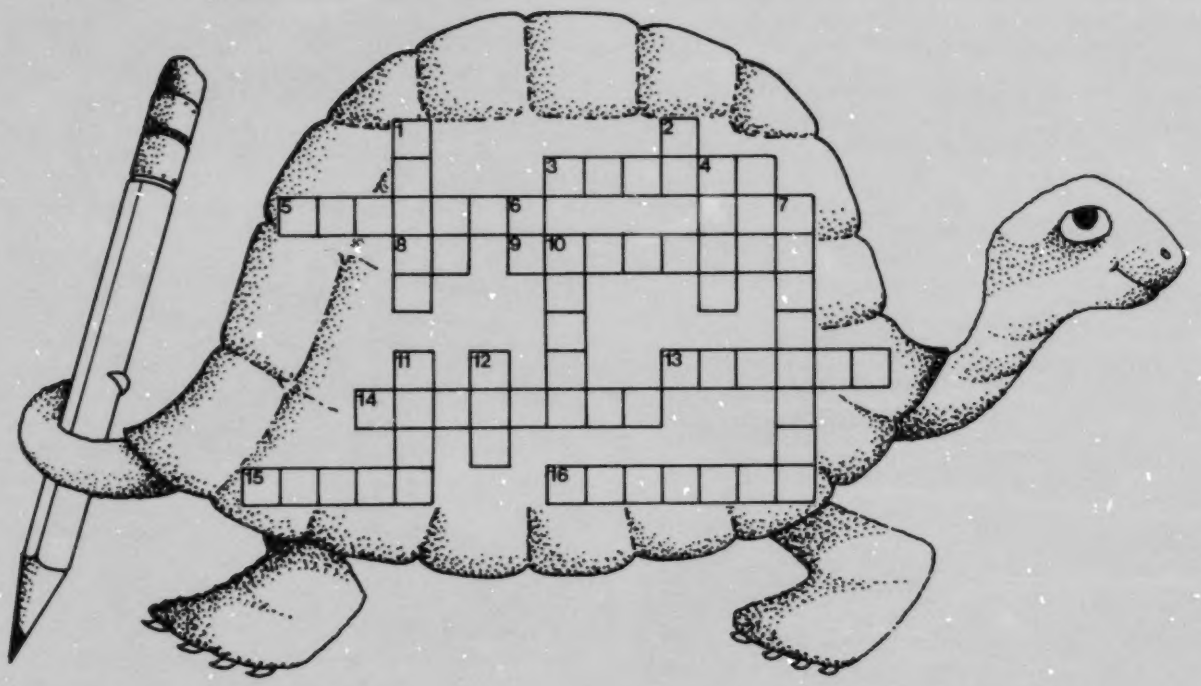
You may want to use adult volunteers to assist students with these introductory skills, but caution volunteers to allow students as much freedom to explore as possible. (See "Volunteers," page 9 .)

Have students complete the Problem Sheet.

*The Atari version of Logo uses a figure that resembles a "real" turtle.

Session 3
Problem Sheet

Name _____
Date _____



ACROSS

3. What we call the figure on the Logo screen that does the drawing.
5. The part of the computer system that shows what you type on the keyboard, and shows the drawings of Turtle.
8. The abbreviated command that moves Turtle forward.
9. The shape of the turtle (except in Atari Logo).
13. The part of the computer that remembers the instructions or commands that you have typed.
14. The monitor, keyboard, CPU, disk drive or cassette recorder are all parts of the _____ system.
15. This marks the end of a command.
16. An order or instruction you give Turtle that has a special meaning in Logo.

DOWN

1. Hold this key down to print the upper part of a two-part key.
2. The abbreviated command to turn Turtle left.
4. The name of the computer language you are learning.
6. The abbreviated command to turn Turtle right.
7. The part of the computer system where you type the commands.
10. The original Logo Turtle was a _____.
11. The place in the center of the screen where Turtle is found when you begin using Logo.
12. The abbreviation for the part of the computer system that does the work of a brain.

WORD BANK

FD	COMMAND	DASH	LEFT	PERIOD	SQUARE
BK	COMPUTER	END	LOGO	ROBOT	TRACKS
RT	CPU	HOME	MEMORY	SHIFT	TRIANGLE
LT	CTRL	KEYBOARD	MONITOR	SPACE	TURTLE

© ICCE 1984. Reproduction for classroom use permitted.

Session 4

*"The first [discussion of observations] deals with the way in which children reconstruct their own movements or changes of position by drawing on their own conceptions of the spatial field. . . . The second deals with the way in which they measure, or show those more primitive kinds of behavior which eventually lead to measurement itself; the actual situation is a kind of game in which the child is asked to make a construction which he recognizes sooner or later as one which demands a degree of precision that only measuring can guarantee."**

Jean Piaget, Barbel Inhelder, Aline Szeminska
The Child's Conception of Geometry

OBJECTIVES: To examine how we change directions by turning, or pivoting, and to introduce the protractor as a tool for measuring the amount of turning.

CONCEPTS/SKILLS: Use of a protractor and conceptual development of an angle.

FACILITIES/MATERIALS: Gym, playground or hallway; for each pair of students: a meter stick, water-base markers or chalk (caution: check marker washability on the floor before using on a large area), damp sponge or paper towel for erasing, large protractor, directions for two different shapes; rolls of newsprint or butcher paper if you prefer that students not write on the floor; Using the Large Protractor wall chart; small protractors for older students; and student copies of the Problem Sheets.

PREPARATION: Masters for the shape directions used in this session are included after the ASSIGNMENT section. Version A is for younger students who will be measuring in centimeters and using large protractors for drawing on the floor. Version B is for older students who will be measuring in millimeters and using small protractors for drawing on their paper. Large Protractors master and Using the Large Protractor wall chart are in the Charts section. Protractor masters need to be duplicated on tagboard or duplicated and mounted on tagboard and cut out and assembled.

PRIMITIVE CHART:

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR

COMMENTS:

Logo is a powerful tool for motivating and studying geometry. Even the youngest student can see that one can make turns other than RIGHT 90 and LEFT 90. Students can see the need to specify larger or smaller Turtle turns.

This session is a combination of formal instruction on angular measurement and continued work with Logo. The work with Logo includes use of commands such as RIGHT 45 and LEFT 30. The student is encouraged to explore—to develop a concrete understanding of angular measurement and its application when working with Turtle.

Two new Logo primitives are introduced in this session. HOME moves Turtle to its initial (center of screen) position, pointed upward. In APPLE, ATARI and MIT, Turtle draws a line during this movement if PENUP has not been specified. (At this stage students have not yet learned PENUP and PENDOWN; in everything they are able to do so far, Turtle is in the PENDOWN position.)

The second primitive clears the screen (CLEAN,

*Reprinted by permission of Humanities Press Inc., Atlantic Highlands, NJ 07716.

CLEARSCREEN [CS] or CLEAR). The CS command in APPLE and ATARI is different from CS in MIT and TI. In APPLE and ATARI, CS puts Turtle into the drawing mode. That is, it clears the screen and returns Turtle to home. In MIT and TI, CS just clears the screen and leaves Turtle in the same position. This is accomplished by the CLEAN command in APPLE and ATARI.

You will want to explain and illustrate the new primitives to your students. Now enough primitives have been introduced, so that some students will have trouble keeping them all in their heads. The value of having a wall chart posted by each computer becomes evident. Students will learn the primitives by repeated use. It is important that they learn to use the chart to supplement their memories.

ACTIVITY:

Part 1

After reviewing the primitives, introduce the new ones for this session. Be sure to add the new primitives to the wall chart.

Part 2

Discuss the word "pivot." Ask a student to locate the word in a dictionary. Discuss the meaning, especially as it is used in basketball, so students are aware of turning without moving from one spot.

Pass out the large protractors, one to each pair of students. Allow time for students to examine them. Call attention to:

- The shape—a half circle with one straight edge.
- The midpoint on the straight edge.
- The double numbering system on the curved edge—0 to 180.

Go to the gym, playground or hallway. Bring the wall chart for reference. Demonstrate and explain:

To measure the amount of turning to the right, hold the protractor in the right hand. Place the protractor on the floor with the straight edge alongside the right foot and the midpoint at the heel of the shoe. Hold up the wall chart to illustrate. Extend the right arm straight out, pointing in the direction you wish to turn. Have the partner make a mark on the floor showing the direction to turn. Emphasize that this mark is not an amount forward but simply indicates the amount of turning to change the direction of movement. This method measures the amount of pivoting and does not include any forward movement. After the partner has made the mark on the floor, again look at the protractor. Begin counting from the zero nearest the toe of the shoe. Stop at the mark on the floor. This number is the amount to turn. The units on the protractor are called degrees, and they measure the amount of turning required to change directions.

Repeat the demonstration on the left side.

Give ample time for students to practice turning varying amounts both right and left, interspersed with FD commands. The concept that turning is separate from the commands for FD and BK may need to be emphasized with younger children.

ASSIGNMENT:

Using meter sticks, protractors and water-base markers, have the students draw these shapes on the floor. Rolls of newsprint or butcher paper may be placed on the floor if you prefer.

1) FD 30 (cm)
RT 90 (degrees)
FD 30
RT 90
FD 30
RT 90
FD 30
RT 90
END

2) FD 40 (cm)
LT 120 (degrees)
FD 40
LT 120
FD 40
LT 120
END

Erase with damp sponges or towels when the tasks are completed.

Give each pair of students at least two cards which contain directions for making shapes using the FD, BK, RT and LT commands. (See Shapes for Session 4 which follow this Assignment section.) Allow ample time for practice using protractors. As a challenge, encourage students to make up a set of directions for another pair of students to draw.

COMPUTER ASSIGNMENT:

Have students sign up for time at the computer. They should take a protractor to the computer to help them guess where the turtle will point when given commands for turning different numbers of degrees.

Have students complete the Problem Sheet for this session at the computer.

Session 4
Shapes—Version A

1

Using degrees and centimeters, draw this shape on the floor or on paper. Cross out each step as you complete it.

```
FD 30
RT 90
FD 30
RT 90
FD 30
RT 90
FD 30
RT 90
END
```

2

Using degrees and centimeters, draw this shape on the floor or on paper. Cross out each step as you complete it.

```
FD 20
LT 90
FD 20
LT 90
FD 20
LT 90
FD 20
LT 90
END
```

3

Using degrees and centimeters, draw this shape on the floor or on paper. Cross out each step as you complete it.

```
FD 30
RT 120
FD 30
RT 120
FD 30
RT 120
END
```

4

Using degrees and centimeters, draw this shape on the floor or on paper. Cross out each step as you complete it.

```
FD 50
LT 120
FD 50
LT 120
FD 50
LT 120
END
```

5

Using degrees and centimeters, draw this shape on the floor or on paper. Cross out each step as you complete it.

```
FD 40
RT 60
FD 40
RT 60
FD 40
RT 60
FD 40
RT 60
FD 40
RT 60
FD 40
RT 60
END
```

6

Using degrees and centimeters, draw this shape on the floor or on paper. Cross out each step as you complete it.

```
FD 30 LT 45
LT 45 FD 30
FD 30 LT 45
LT 45 END
FD 30
LT 45
FD 30
LT 45
FD 30
LT 45
FD 30
LT 45
FD 30
```

© ICCE 1984. Reproduction for classroom use permitted.

Session 4
Shapes—Version B

1

Using degrees and millimeters, draw this shape on your paper. Cross out each step as you complete it.

```
FD 30
RT 90
FD 30
RT 90
FD 30
RT 90
FD 30
RT 90
END
```

2

Using degrees and millimeters, draw this shape on your paper. Cross out each step as you complete it.

```
FD 20
LT 90
FD 20
LT 90
FD 20
LT 90
FD 20
LT 90
END
```

3

Using degrees and millimeters, draw this shape on your paper. Cross out each step as you complete it.

```
FD 30
RT 120
FD 30
RT 120
FD 30
RT 120
END
```

4

Using degrees and millimeters, draw this shape on your paper. Cross out each step as you complete it.

```
FD 50
LT 120
FD 50
LT 120
FD 50
LT 120
END
```

5

Using degrees and millimeters, draw this shape on your paper. Cross out each step as you complete it.

```
FD 40
RT 60
FD 40
RT 60
FD 40
RT 60
FD 40
RT 60
FD 40
RT 60
FD 40
RT 60
END
```

6

Using degrees and millimeters, draw this shape on your paper. Cross out each step as you complete it.

```
FD 30 LT 45
LT 45 FD 30
FD 30 LT 45
LT 45 END
FD 30
LT 45
FD 30
LT 45
FD 30
LT 45
FD 30
LT 45
FD 30
```

© ICCE 1984. Reproduction for classroom use permitted.

Session 4 **Name** _____

Problem Sheet, Page 1 **Date** _____

(Use at the computer.)

1. Pretend you are in the draw mode with a clear screen and Turtle in the center facing up. You give Turtle the following commands:

RT 90
FD 50
LT 90
FD 50
HOME

What picture do you think Turtle will draw? Draw that picture here before you enter it on the computer.

Now, try it on the computer and sketch that picture here.

2. Beginning with a clear screen and Turtle in the center facing up, give Turtle the following commands:

FD 50
LT 45
FD 70
HOME

Draw a picture of the screen and explain in your own words what HOME does.

© ICCE 1984. Reproduction for classroom use permitted.

Session 4
Problem Sheet, Page 2
(Use at the computer.)

Name _____
Date _____

3. Which primitive clears the screen? _____

4. How much is the total turning if Turtle leaves home and returns again?

   Start solving this problem by clearing the screen and facing Turtle straight up. Now take Turtle for a little walk. Tell it to go forward some steps and then turn right some. Write down on a piece of paper how much it turned. Then tell it to go forward some more steps and to turn right some more. Write down that amount of turning and add it to the first turn. Keep doing that until finally Turtle is facing straight up again in the beginning position.

   How much turning did Turtle do to turn all the way around? _____

5. What shapes can you make if you tell Turtle to go forward some amount and then turn right some amount, and keep repeating those two commands with the same amounts?

   Example:
   FD 40
   RT 120
   FD 40
   RT 120
   FD 40
   RT 120

   How many total degrees did Turtle turn? _____

   Look at that shape on the screen. Sketch it here:

   Example:
   FD 40
   LT 144
   FD 40
   LT 144
   FD 40
   LT 144
   FD 40
   LT 144
   FD 40
   LT 144

   How many total degrees did Turtle turn? _____

   Look at that shape on the screen. Sketch it here:

© ICCE 1984. Reproduction for classroom use permitted.

Session 4
Problem Sheet, Page 3
(Use at the computer.)

Name _____
Date _____

6. Make a figure of your own. Start by facing Turtle straight up in the center of the screen. Write down what you tell Turtle to do. Look at the shape you have made on the screen and sketch that shape here:

FD_____

© ICCE 1984. Reproduction for classroom use permitted.

Session 5

*"There are situations where this style of thinking [step-by-step, literal, mechanical fashion of computer programs] is appropriate and useful. . . . By deliberately learning to imitate mechanical thinking, the learner becomes able to articulate what mechanical thinking is and what it is not. The exercise can lead to greater confidence about the ability to choose a cognitive style that suits the problem. Analysis of 'mechanical thinking' and how it is different from other kinds and practice with problem analysis can result in a new degree of intellectual sophistication. . . . giving children the opportunity to choose one style or another provides an opportunity to develop the skill necessary to choose between styles."**

Seymour Papert
Mindstorms: Children, Computers, and Powerful Ideas

[Note: Younger students need repetition to internalize concepts and develop an intuitive sense for spatial relationships. For older students, some sessions may cover concepts/skills which they may already have mastered. Such sessions could easily change to focus on proper keyboarding techniques.]

OBJECTIVES: To introduce the concept of writing a set of directions, to reinforce the concept of measuring the amount of turning, and to provide practice in using the protractor.

CONCEPTS/SKILLS: Sequencing, turning and use of a protractor.

FACILITIES/MATERIALS: Locations on the school grounds; large protractors and blank 5x8 cards for each pair of students; and student copies of the Problem Sheet.

PRIMITIVE CHART:

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR

COMMENTS:

This session has two purposes. The first is to serve as a pause, a time for consolidation and continued practice. By now all of your students should feel confident in their ability to enter the **draw mode**, type in a short sequence of commands, and correct typing errors that are detected before a command is executed. Your students should use computer vocabulary (such as keyboard, screen and Turtle) correctly. They should have skill in reading simple sets of Logo commands and predicting what Turtle will do when executing these commands. They should have some skill in writing a sequence of commands to produce a picture of their own design.

The second purpose of this session is to introduce the idea of a procedure. In Logo a procedure consists of a title line, a body and the word END.

*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

```
TO SQUARE
FD 50
RT 90
FD 50
RT 90
FD 50
RT 90
FD 50
RT 90
END
```

The first line is the title line. It contains the word TO followed by the name of the procedure being defined. The body consists of all subsequent lines before the END line.

In Logo, a procedure defines a new command for Turtle (the name of the procedure), and that command can be used just like any primitive is used. That is, defining a procedure extends the Logo vocabulary. This capability in Logo, as well as the general idea of procedure, is a powerful idea. A substantial portion of the remainder of this book is devoted to learning to write and use Logo procedures. This session introduces procedures by using some of the procedure vocabulary with sets of directions that students write to particular locations.

The idea of procedure occurs in many areas other than Logo. A recipe is a procedure for cooking. Your students know procedures for adding a column of numbers and for tying their shoes. You may want to spend time on non-Logo procedures. A number of examples are given in *Computer Literacy Activities for Elementary and Middle School Students.**

ACTIVITY:

Pass out the large protractors and discuss the previous activity. Review how to use the protractors.

Have students work in pairs. Explain that they are to write a set of directions to go to a place of their choosing. This place should be either on the playground or in the school building. A new name for the set of directions is a "procedure." In their procedure they may use FD, BK, RT and LT, but the commands FD and BK must have numbers telling how many steps. The commands RT and LT must include a number telling how many degrees to turn. Tell students to begin their cards by writing where they are going.

Example: TO BIKE RACK or TO FRONT DOOR

Write the directions; tell when the procedure ends by using the word END.

```
Example: TO BIKE RACK
         FD 20          (number of steps)
         RT 40          (number of degrees)
         FD 5
         RT 40
         FD 10
         LT 45
         FD 6
         END
```

ASSIGNMENT:

Have students sign up for one or two slots on the Time-On chart. Students should practice moving the turtle on the screen. (This practice will help them internalize the primitives and will reinforce the concept of turning and the skill of using degrees to measure the amount of turning. Students will also be practicing keyboarding skills.

Have students complete the Problem Sheet for this session.

*International Council for Computers in Education (ICCE), 1983.

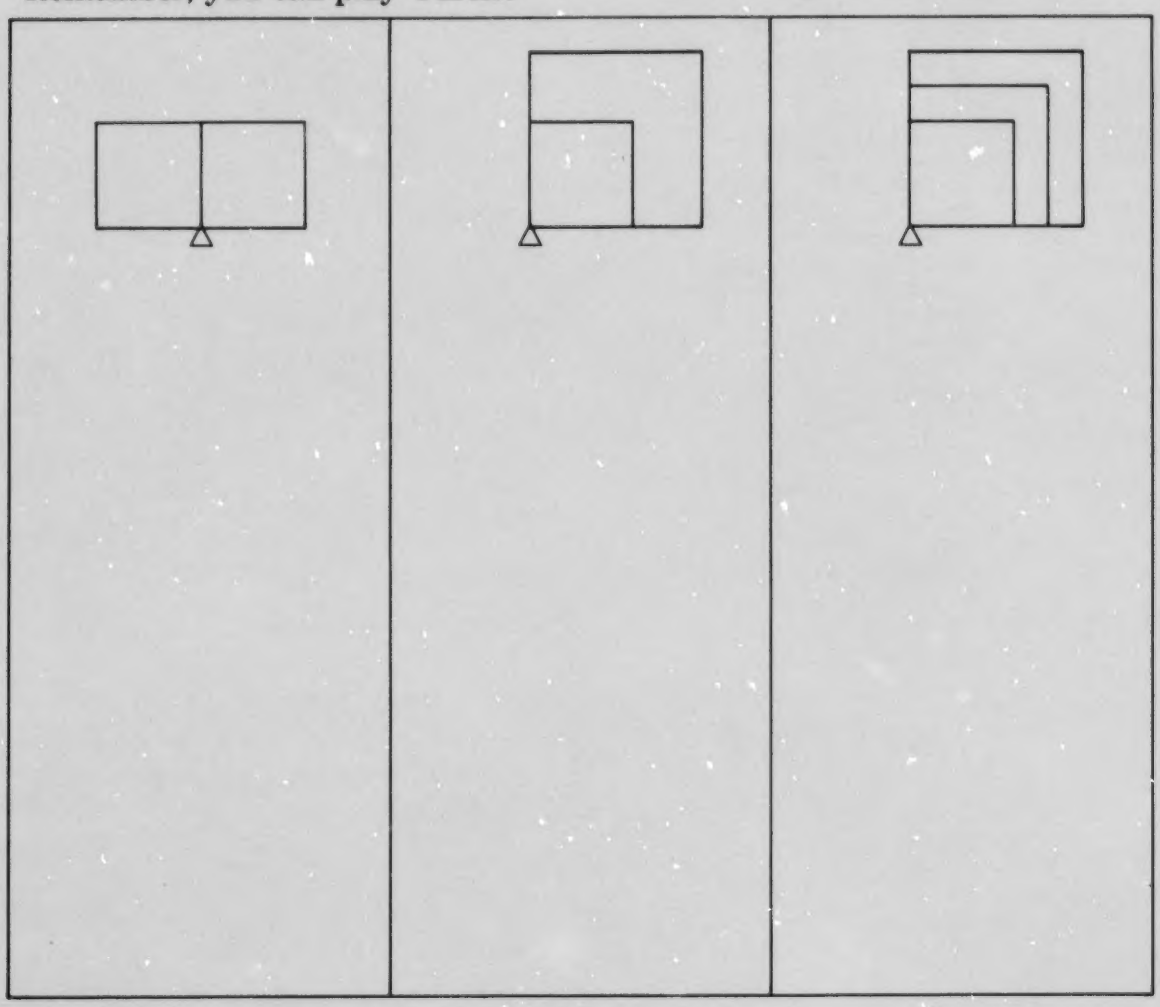
Session 5
Problem Sheet

Name _____
Date _____

READ ALL OF THIS PAGE *BEFORE* YOUR COMPUTER "TIME-ON" PERIOD.

1) Write a set of directions (procedure) telling Turtle how to draw ONE of the following designs. It does not have to be the same size as the examples. You may choose your own number of turtle steps. You may write the directions and then try it on the computer, or you may work directly on the computer.

Remember, you can play Turtle!



2) Write a set of directions for a design you have made on the computer screen. Use the back of this sheet to write the procedure and sketch the design.

© ICCE 1984. Reproduction for classroom use permitted.

Session 6

*"Geometrical intuition is essentially active in character . . . and if the action itself is inadequate, intuition breaks down. This is evident from all the experiments described . . . all the forms of spatial intuition which have been studied depend upon actions."**

Jean Piaget and Barbel Inhelder
The Child's Conception of Space

OBJECTIVES: To reinforce measuring "the amount of turning" in an angle and to review metric measurement relative to Turtle steps.

CONCEPTS/SKILLS: Metric measurement: m, cm, mm.

FACILITIES/MATERIALS: Overhead. For each student: large protractors, meter sticks, metric rulers and graph paper (see Charts section). Overhead metric ruler or 6 cm transparent strip marked in cm and mm; student copies of the Problem Sheet.

PRIMITIVE CHART:

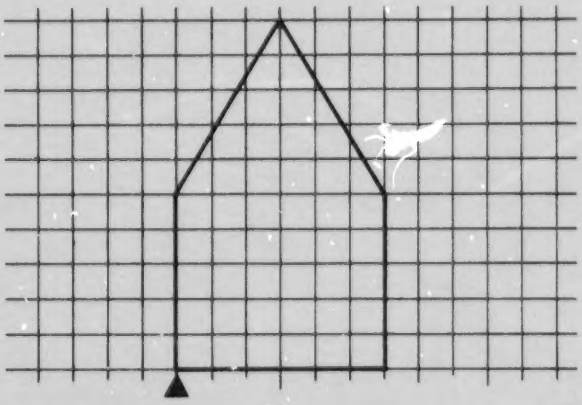
APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR

COMMENTS:

A Logo computer system can be used as a toy, with a student making up patterns while seated at the keyboard. But eventually students will (perhaps with a little encouragement from you) begin to conceive of patterns they would like to produce and go to the machine with a definite goal in mind.

Student desire to direct Turtle to draw specific patterns can serve as a motivation for studying linear measurement, angular measurement, and use of graph paper. This session concentrates on linear measure using the metric system. Metric graph paper is used and related to developing patterns to be drawn using Logo.

As a review and to motivate this session, you might put on the chalkboard or an overhead projector two house-shaped patterns as below.



*Reprinted by permission of Humanities Press, Inc., Atlantic Highlands, NJ 07716.

Work with the class to measure the turns (angles) needed to make this shape. Discuss how the graph paper aids in making a neat, carefully drawn pattern of a house. Discuss measuring the lengths of the sides in the drawing. For example, in the drawing on graph paper, the left side is five units long. Determine the other lengths. Have students walk this house pattern, using a very short step as a unit of measure. Then have them walk the same pattern using a very long step as a unit of measure. Students should see that the shape of the pattern remains the same regardless of the unit of measure (length of a step), but the size of the pattern varies.

ACTIVITY:

Part 1

Pass out the meter sticks. Discuss the mm and cm markings as you illustrate with the overhead ruler or transparent strip on the overhead projector. (Do not call attention to dm at this time.) After students have examined millimeter and centimeter markings, ask questions, helping students count by 10's if necessary.

- How many mm in 1 cm? (10)
- How many mm in 3 cm? (30)
- How many mm in 9 cm? (90)
- How many mm in 10 cm? (100)
- How many mm in 20 cm? (200)
- How many mm in 30 cm? (300)
- How many cm in 1 m? (100)

Let students measure objects in the room such as desks, windows and books.

Compare and discuss the meter sticks and metric rulers.

Part 2

On the overhead, review how to draw lines using a ruler. (The end of the ruler must be on the endpoint of line; view from above—not from the side.) Allow practice time.

Tell students they have probably discovered by now that FD 1 isn't very far forward. For Turtle to make designs, they need to direct it forward a number greater than 1 or 2. Just as people can go forward more than 1 or 2 steps at a time, so can Turtle.

Pass out the graph paper (where the length of 5 squares = 1 cm). Discuss using the paper as a model of the screen where Turtle "lives." Carefully explain the scale used on this graph paper, and how it is to be used relative to Turtle steps.

Scale: 1 mm = the length of one Turtle step

Tell students that we know there are 10 mm in 1 cm, but this paper shows only 5 parts to 1 cm. Graph paper that shows 10 parts to 1 cm has the lines so close together that the lines are hard to see and draw on. Since there are only 5 parts in 1 cm on this paper, the length of one of the 5 parts is the length of 2 Turtle steps, and the length of 10 Turtle steps is 1 cm.

2 Turtle steps = the length of the small square (2 mm)
10 Turtle steps = the length of the large square (1 cm)

Ask students: If 2 Turtle steps = the length of 1 small square on the paper, how many Turtle steps are there in a line:

- 1 cm long? (10)
- 2 cm long? (20)
- 3 cm long? (30)
- 4 cm long? (40)
- 5 cm long? (50)
- 10 cm long? (100)
- 1 m long? (1000)

Using the chalkboard:

- Draw a 10 mm (1 cm) line and tell students that this line represents 10 Turtle steps.
- Draw a 20 mm (2 cm) line and tell students that this line represents 20 Turtle steps.

ASSIGNMENT:

Have students:

a) Draw on a piece of paper lines with the following lengths:
   1) 20 mm
   2) 30 mm
   3) 50 mm
   4) 100 mm
   5) a line to represent 20 Turtle steps
   6) a line to represent 30 Turtle steps
   7) a line to represent 50 Turtle steps
   8) FD 20
   9) FD 30
   10) FD 100

b) Sign up for computer time. Have each student draw on a piece of graph paper a pattern that they can draw using Logo. (Remember, the pattern must consist of a single, connected line.) Each student is to write Logo commands to draw his/her pattern. Then have students try out their directions on the computer during their computer time.

c) Complete the Problem Sheet for this session.

Session 6
Problem Sheet

Name _____
Date _____

1) Can you find out how many Turtle steps from HOME to each edge of the screen? (Make a guess and then try it on the computer.)

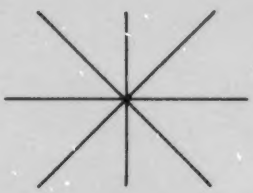
HOME to top?_____ HOME to bottom?_____

HOME to left edge? _____ HOME to right edge? _____

2) Try this at the computer. You may work in pairs or in small groups. Remember to clear the screen before trying the following:

```
HOME
FD 80
BK 80
RT 15

FD 70
BK 70
RT 15

FD 60
BK 60
RT 15

FD 50
BK 50
RT 15

FD 40
BK 40
RT 15

FD 30
BK 30
RT 15

FD 20
BK 20
RT 15

END
```

3) Can you make Turtle draw one of these? Write a set of directions for one of them.



© ICCE 1984. Reproduction for classroom use permitted.

Session 7

*"One of the most widespread representations of the idea of angle in the lives of contemporary Americans is in navigation. Many millions navigate boats or airplanes or read maps. . . . We have stressed the fact that using the Turtle as metaphorical carrier for the idea of angle connects it firmly to body geometry. We have called this body syntonicity. Here we see a cultural syntonicity: The Turtle connects the idea of angle to navigation, activity firmly and positively rooted in the extraschool culture of many children."**

Seymour Papert
Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVES: To introduce and give practice in using a small, desk-size protractor and to introduce the concept of "angle" and relate it to the amount of turning done in changing directions in movement.

CONCEPTS/SKILLS: Directionality, developing an understanding of angles in relation to self, and use of protractors.

FACILITIES/MATERIALS: Overhead. For each student: small protractor, large protractor, metric ruler and unlined paper. Small paper triangle to represent Turtle on the overhead; and student copies of Practice Sheets, Problem Sheet and Challenger Sheet.

PREPARATION: Locate points A, B and C on the overhead so that when the rays AB and AC are drawn, they will form a 40° angle (see Demo #1 within the ACTIVITY section).

PRIMITIVE CHART:

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR

COMMENTS:

The previous session emphasized linear measurement using the metric system. By now you have illustrated use of a large protractor several times. Students have had an opportunity to use a large protractor as an aid to walking sets of directions containing turns.

If your students have had previous instruction in using a small protractor, then this session can serve as a review of that topic. If the small protractor is new for your students, you will need to supplement the material of this session with additional instruction.

ACTIVITY:

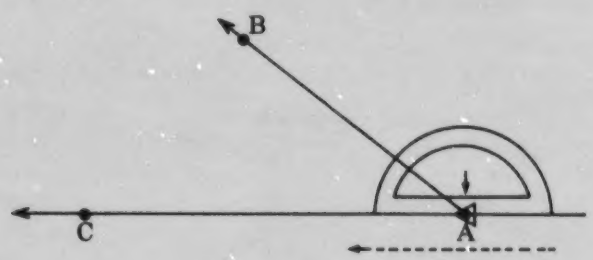
Depending on the developmental level of your students, you may want to have them walk out the first demonstration using a larger scale (60 cm instead of 6 cm) before passing out the small protractors. This self-involvement in the activity should provide a smooth transfer to the desk-size problem.

*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

Pass out the small protractors and let students compare them to the large protractors. Call attention to the double numbering system. Emphasize that in Turtle geometry, the straight side of the protractor is placed along the direction of movement (illustrate), and that one must then look ahead, find the zero and use that set of numbers. Point out the midpoint on the straight edge of the small protractor.

Demonstrate the above procedure on an overhead transparency, as follows:

DEMO #1.



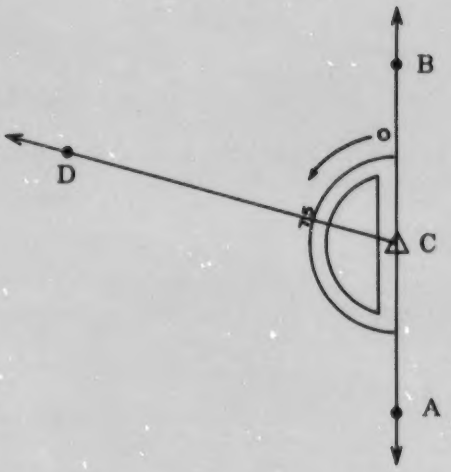
Draw a line about 6 cm long. Tell students this is the path Turtle is walking. Turtle is at point A, heading toward point C. (Illustrate by placing the cut-out triangle over A pointing to C.) Turtle wants to go to point B from point A. (Turn the triangle to head toward B. Remove the triangle and use a ruler to draw a line from A through B.)

Turtle must turn—change its direction—before it can move forward. (Illustrate the turn again with the triangle.) How much does Turtle need to turn? We can use a protractor to find out. Point B is on Turtle's right side, which we will call the right side of the line. Just as the large protractor was placed on your right side, the protractor must be on the right side of the line with the straight edge next to the line. The midpoint of the protractor must point to A.

Look straight ahead toward point C (at the end of the protractor) to find the zero. Use that set of numbers. Count on the curved edge of the protractor until you reach the point where line AB passes under the protractor. This number, 40°, is the amount Turtle must turn from facing point C to face toward point B. Now Turtle can move forward to point B.

If you want Turtle to turn this amount, you tell Turtle: RT 40. This will turn Turtle 40° to its right.

DEMO #2.



Give each student a sheet of unlined paper and have them do this exercise as you talk them through it.

Draw line AB on the overhead transparency. Have students do the same on their paper. Label points A, B and C. (See Demo #2.)

Tell students that Turtle is at point C, facing point B. The commands to Turtle are:

LT 75
FD 50

Place the protractor on Turtle's left side, since Turtle will turn left. Place the midpoint arrow at point C, with the straight edge of the protractor along the left side of line AB. Look straight ahead toward point B, since that is the direction Turtle is heading.

Find the zero to know which set of numbers to use. Use this set to count from the zero up to 75. Make a point by the 75° mark on the protractor. Remove the protractor. Now use a ruler to draw a line from point C through the new point you made at the 75° mark. The line should be 50 mm (5 cm) long since Turtle steps (FD 50) would be about 50 mm using the graph paper.

Tell students that when two lines (or line segments, if they have already been exposed to that term) such as $\overline{CB}$ and $\overline{CD}$ go in different directions from the same point C, the lines form an *angle*. This angle can be measured by using the protractor. The angle just made is called angle BCD or angle DCB. It can be written $\sphericalangle$ BCD or $\sphericalangle$ DCB.

Optional Activity:

Here is an idea for an activity to strengthen social studies and map skills while providing practice in multiplication facts and in the internalization of the Logo primitives FD, BK, RT and LT.

You and your students might design the game as an independent activity for two students to play at their desk or on a table; or it could be designed as a whole class contest, using a large pull-down map, with each student having one turn each day for an agreed-upon number of days.

Use a world map of an appropriate size showing longitude and latitude lines. Provide a pair of dice, one marked 3, 4, 5, 6, 7, 8 and the other marked 7, 8, 9, 10, 11, 12; a spinner marked from 0 to 360 (the total number of degrees in a circle); a ruler and protractor; and a marker for each player. (The marker needs an indication of heading—for example, a small paper airplane.)

The object of the game is to see how many countries a player can visit in the agreed-upon time, and still be the first to complete a trip around the world (one side of the map to the other).

Each airplane begins at a specified place and heading on one side of the map. Players, in turn, either roll the dice **or** spin the spinner. The dice are rolled if the player wants to move FORWARD or BACK. The numbers rolled are multiplied together and the airplane then moves ahead that many **hundreds** of miles (3*8=24 hundred miles), using the distance key on the map.

To change direction, the player must spin the spinner and turn that many degrees from the present position either RIGHT or LEFT (no FORWARD or BACK movement).

Players can keep score by listing the countries they "visit" (land on). Encourage students to discuss their strategies. (For a large map, yarn might be used to keep track of each student's journey, and provide useful information on different strategies!)

BON VOYAGE!

ASSIGNMENT:

1) Have students practice drawing some angles. Encourage students to share and help each other.

2) Complete the Practice Sheet.

3) Complete the Problem Sheet.

4) Complete the Challenger Sheet (if appropriate).

5) Sign up for computer time.

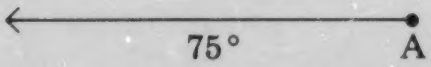
Session 7
Practice Sheet

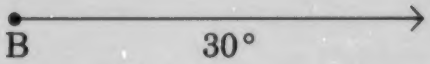
Name _____
Date _____

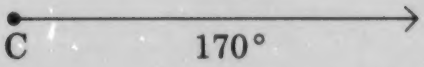
Can you draw angles of the following sizes?

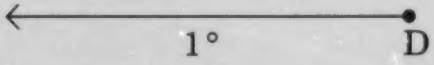
1. 75°
2. 30°
3. 170°
4. 1°
5. 180°

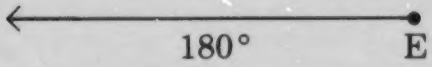
To draw the angles, use the lines drawn below. Place the straight edge of the protractor along the line for each angle, with the midpoint of the protractor on the lettered point.











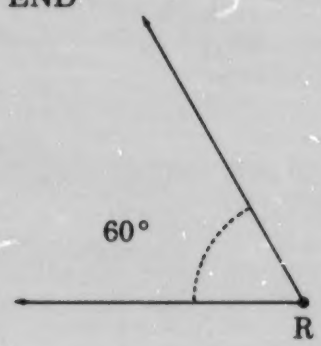
Can Turtle draw these angles? Try them at the computer.

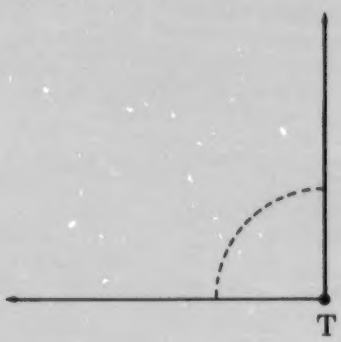
© ICCE 1984. Reproduction for classroom use permitted.

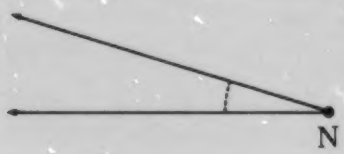
Session 7 **Name** _____

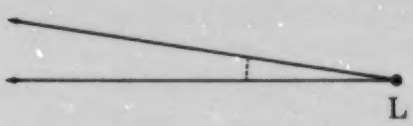
Problem Sheet, Page 1 **Date** _____

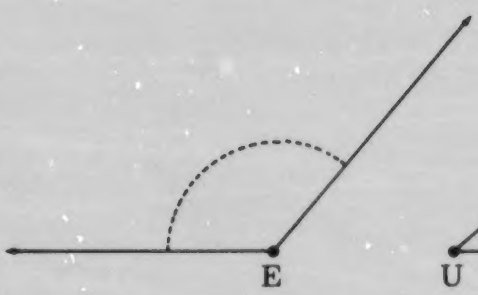
TO FILL IN SECRET MESSAGE
MEASURE ANGLE
MARK DEGREES ON ANGLE
MATCH ANGLE MEASUREMENT BELOW
WRITE TURNING POINT LETTER ON LINE ABOVE MATCHING DEGREE NUMBER
REPEAT UNTIL ALL SPACES ARE FILLED
READ SENTENCE
TELL TURTLE NEW PRIMITIVE
END

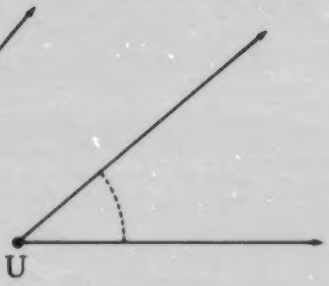


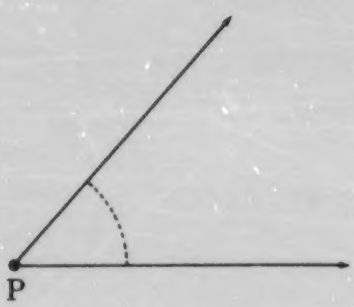












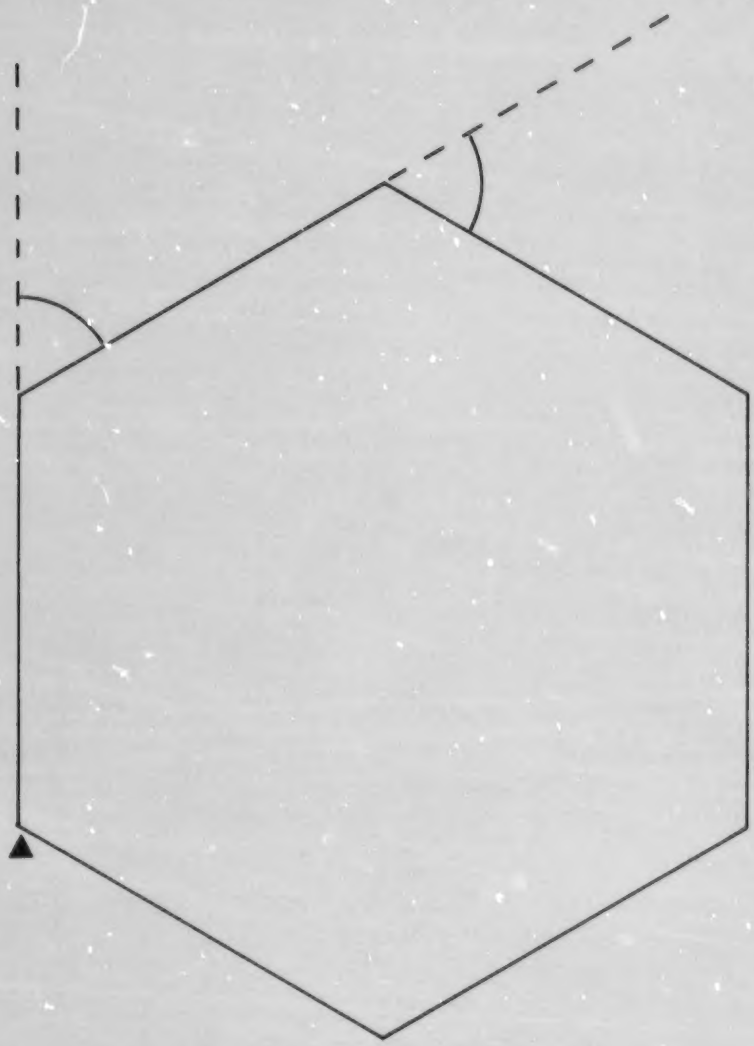
					R						
90	130	10	10		90	40	60	90	10	130	
									50	130	20	40	50

© ICCE 1984. Reproduction for classroom use permitted.

Session 7 **Name** _____

Problem Sheet, Page 2 **Date** _____

The figure below is a hexagon. It has six equal sides and six equal angles. Measure the angle Turtle needs to turn after drawing one side of the hexagon. Write Logo commands to draw a hexagon. Try them out on a computer.



© ICCE 1984. Reproduction for classroom use permitted.

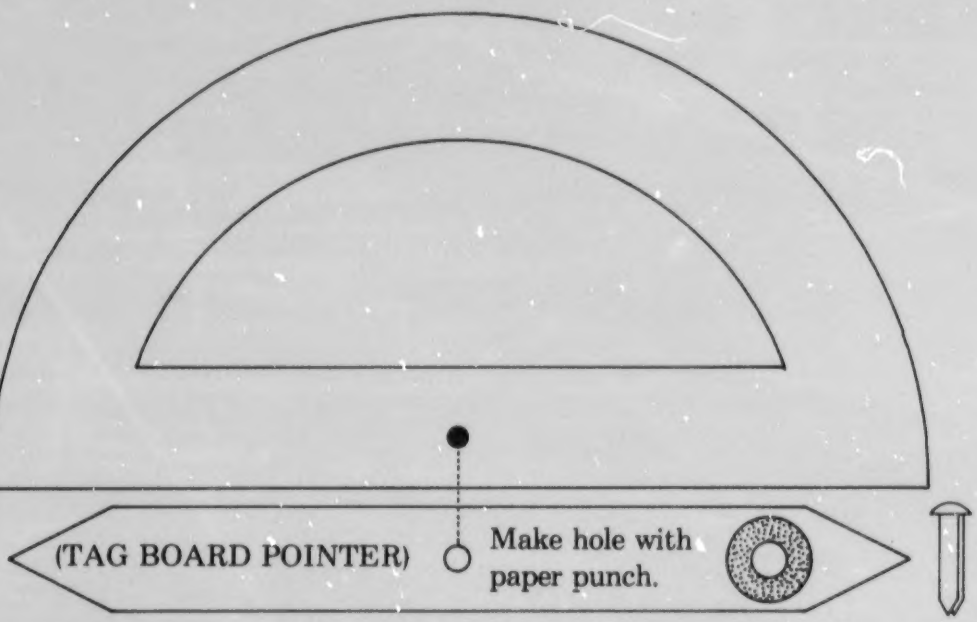
Session 7 **Name** _____

Challenger Sheet **Date** _____

1) Make an angle measurement tool:

Materials: Protractor, tagboard, small round-head paper fastener, penny or washer.

To assemble: Cut a pointer out of tagboard and fasten to the protractor with the paper fastener. Most protractors have a small hole at the middle point of the straight edge. If this hole is not large enough for the paper fastener, it must be enlarged using a small drill so the paper fastener will turn freely. Tape or glue the penny or washer to one end of the pointer to act as a weight.



2) To use: The angle measurement tool is held with the straight edge of the protractor parallel to the line of angle being measured. (Putting the edge of the protractor directly on or under the surface restricts the movement of the pointer.) Next, rotate the pointer to line up with the other line of the angle.

3) Students can see how many different angles they can find in the room, library or playground. Look for handrails on stairways, playground slides, porch roofs, playground fencing, etc.

Develop a project for angular measurement of a ramp, testing the length of distance traveled by toy cars at various beginning heights.

Use the problem-solving strategy of making diagrams and tables to answer:

- What distance does the car travel at various angles?
- What predictions can be made?
- What conclusions can be drawn?

© ICCE 1984. Reproduction for classroom use permitted.

Session 8

*"Angles exist only in a context of straight lines, and therefore they are not ready-given in nature, any more than straight lines themselves. Straight lines are relations born out of particular ways of organizing perception, or at a higher level, out of the action of aiming or maintaining a given direction. They are not abstracted from the object; they are added to it as a result of these actions. Angles, likewise, are constructed in the course of action before giving rise to operations; they are not abstracted from the object like heat or weight."**

Jean Piaget, Barbel Inhelder, Aline Szeminska
The Child's Conception of Geometry

OBJECTIVES: To introduce the Logo primitives PENUP and PENDOWN, and the primitive for erasing; to help students develop concepts of distance in Turtle steps and in angles by the amount of turning; and to provide practice for internalizing the Logo primitives.

CONCEPTS/SKILLS: Estimating distance and angle measurement.

FACILITIES/MATERIALS: Computer with Logo cartridge, disk or module, and disk or cassette containing games for Session 8; overhead, chalkboard or video display of computer; display copy of Color Chart; and student copies of the Problem Sheet.

PREPARATION: Enter the procedures for the game into the computer and save them on a disk or cassette. (Procedures are listed after the ASSIGNMENT section.) Be sure to try all games and procedures before using them with your students. Bugs can creep in unexpectedly! After the game for this session has been loaded into the computer, students can activate the game by typing HELP.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)

COMMENTS:

Using the Logo commands studied so far, students can only draw connected figures. They cannot, for example, draw a robot face. See Figure 1.

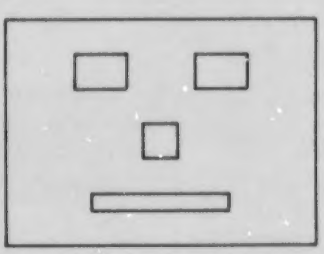


Figure 1.

*Reprinted by permission of Humanities Press Inc., Atlantic Highlands, NJ 07716.

This robot face consists of five distinct pieces.

The PENUP and PENDOWN commands make it possible to draw a pattern consisting of two or more distinct pieces. Now is a good time to begin to emphasize problem segmentation. In the robot face, the eyes and the nose are each a small square. The overall drawing task involves drawing one large square, three small squares and a line segment. You can think of this task as a sequence of five tasks:

Task 1:
A. Position Turtle at starting point for first piece of robot face.
B. Draw first piece of robot face.

Task 2:
A. Position Turtle at starting point for second piece of robot face.
B. Draw second piece of robot face.

Task 3:
Etc.

In the above tasks the initial positioning of Turtle generally involves a PENUP, movement to a specified starting point and a PENDOWN.

Erasing in Turtle graphics consists of redrawing the line segment to be erased, but using the screen's background color. For example, suppose the background color is black and a white line segment has been drawn. Redrawing this line segment in black makes it indistinguishable from the background. That is, even though Turtle has drawn it twice, the resulting line segment is not visible on the screen.

Some versions of Logo have a PENERASE (PE) primitive command. Others require setting the pen color to the background color in order to erase. Check the PRIMITIVE CHART and then your reference manual for details. For example, if your students are using Atari Logo, they will need to learn the PE primitive to erase a line segment. They will also want to learn the SETPC command (set pen color) so they can change the pen color. This allows drawing in different colors if a color display screen is being used.

The last line added to the Primitive Chart in this session is a primitive command to specify a background color. Students will enjoy experimenting with the pen color and background color commands —especially if they are using a color display screen.

ACTIVITY:

Tell students (if they have not already discovered from Session 7 Problem Sheet) that Turtle knows the word PENUP (PU)—and also PENDOWN (PD).

PENUP (PU): Lets Turtle move without drawing any lines.

PENDOWN (PD): Lets Turtle draw with its pen again.

Turtle knows a primitive for erasing, too.

Note: For MIT and TRS, teachers should use their manuals to find the number of the default color of the background and pen color.

APPLE: PENERASE (or abbreviate PE)

ATARI: PE (The abbreviation PE is the primitive in Atari Logo.)

MIT: PENCOLOR n, where n=current background color number (or abbreviate PC n)*

TI: PENERASE (or abbreviate PE)

TRS: PENCOLOR n, where n=current background color number (or abbreviate PC n)

After erasing, Turtle's pen must be made ready to draw again. This is done by use of the following:

APPLE: PENDOWN (PD)

ATARI: PENDOWN (PD)

MIT: PENCOLOR n (PC n), where n=drawing color Turtle had been using

TI: PENDOWN (PD)

TRS: PENCOLOR n (PC n), where n=drawing color Turtle had been using

If you are using a color monitor, students will probably want to be able to change the pen color and background color. Introduce the primitives for changing the pen and background color and add all the new primitives to the Primitive Commands wall chart. Display the Color wall chart by the computer.

APPLE: SETPC ___ (color number) 0-5
SETBG ___ (color number) 0-5

ATARI: SETPC ___ ___ (pen number color number) 0-2 0-127
SETBG ___ (color number) 0-127

MIT: Apple computer:
PENCOLOR ___ (color number) 0-5 or
PC ___ (color number)
BACKGROUND ___ (color number) 0-5 or BG ___ (color number)

Commodore 64:
PENCOLOR ___ (color number) 0-15 or PC ___ (color number)
BACKGROUND ___ (color number)0-15 or BG ___ (color number)

TI: SETCOLOR ___ (color number) 0-15 or SC ___ (color number)
COLORBACKGROUND ___ (color number) 0-15 or CB ___ (color number)

*Commodore 64 also uses PENERASE (PE) or PC −1.

TRS: PENCOLOR ___ (color number) 0-3 in either COLORSET 0 or CCLORSET 1 or PC ___ (color number)
BACKGROUND ___ (color number) 0-3 in either COLORSET 0 or COLORSET 1 or BG ___ (color number)

Tell students that this week they are going to have a contest using a game on the computer. This game will give them an opportunity to practice estimating distances and using the primitives for erasing. Illustrate on the overhead, chalkboard or screen. The object of the game is to put Turtle in the box in the upper left corner of the screen using as few commands as possible. Turtle's path cannot touch any of the other shapes on the screen—it must try to go around any obstacles. If a shape is touched, the part of the line touching the obstacle must be erased using the erase command. The student who gets Turtle into the box using the fewest commands is the winner. Each command must be executed before a new command can be entered. (Each command is counted as one move in the scorekeeping procedure.)

Three students at a time may work at the computer. One student is the player, one keeps a record of the commands, and the third is the timekeeper.

Measuring instruments are not allowed. This is an estimating game, designed to help students develop their estimating skills as well as to provide practice in using the erase command.

Since reflective thinkers may need more time per move (though perhaps fewer moves) than spontaneous thinkers, a time limit should *not* be set on individual moves. An overall time limit per game will be necessary.

Add the new primitives to the Primitive Commands wall chart.

ASSIGNMENT:

Have students complete the Problem Sheet, perhaps while other students are playing the game at the computer.

Have students sign up for computer time to experiment with the pen color and background color commands.

Session 8
Problem Sheet

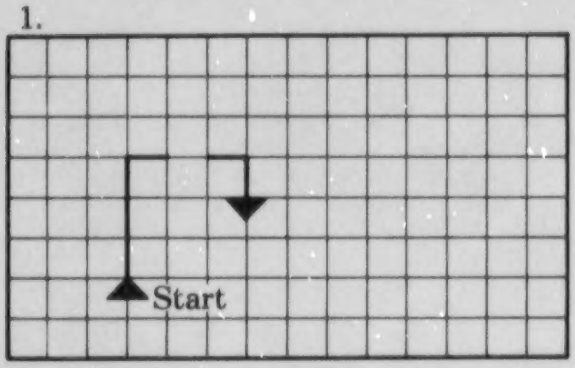
Name _____
Date _____

Can you write a set of directions telling Turtle how to draw any TWO of these shapes?

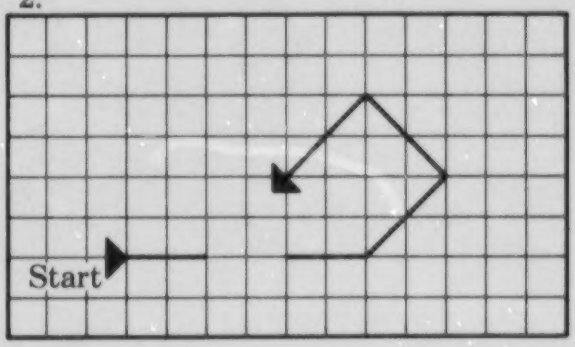
The length of one square on the paper = 10 Turtle steps. To move Turtle a distance on the paper that is one square long, write FD 10. To move Turtle a distance on the paper that is three squares, write FD 30. Remember to use the new primitives PU and PD. You may use your protractor to measure in degrees the amount Turtle turns.

Write your directions on a separate sheet of paper to turn in with this page.

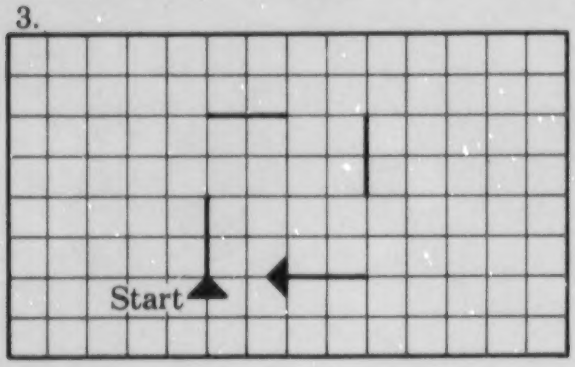
1.



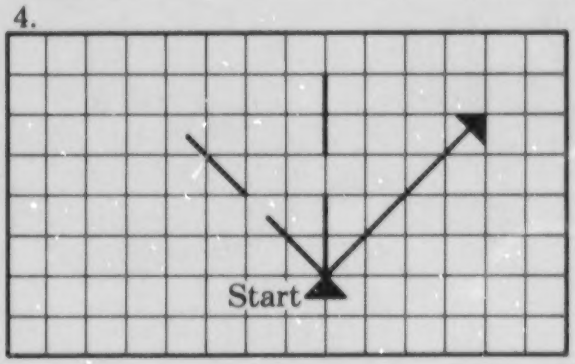
2.



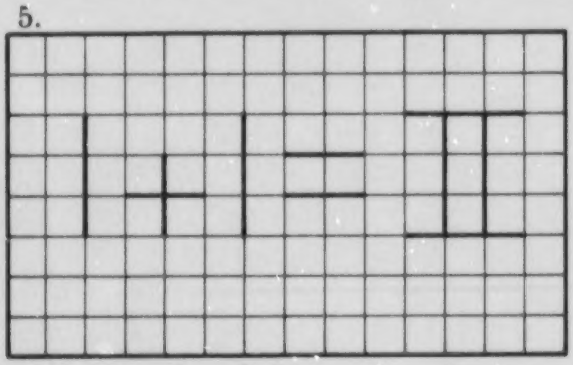
3.



4.



5.



Start and end where you choose.

© ICCE 1984. Reproduction for classroom use permitted.

Session 8
Procedures for Game, Page 1
APPLE

This game provides practice in estimating distances and degrees and in using the primitive for erasing. The object of the game is to put Turtle in the box in the upper left corner of the screen using as few commands as possible (FD, BK, RT or LT). If a shape is touched, the part of the line touching the obstacle must be erased using the erase command. Each command must be executed before a new command can be entered. Each command is counted as one move in the scorekeeping process.

Enter the following procedures into the computer. Try the game to make sure it runs correctly. Game 2 is a slightly more difficult version of Game 1. Save the procedures under the name GAME8 or something suitable.

You will need to load the game procedures into the computer using the name you saved them under. Then students activate the game by typing HELP.

```
TO HELP
TEXTSCREEN CLEARTEXT SETCURSOR [ 0 0 ]
PRINT [ .....ESTIMATING..... ]*
SETCURSOR [ 0 7 ] PRINT [ TO PLAY GAME1,
  PRESS 1. ]
SETCURSOR [ 0 10 ] PRINT [ TO PLAY GAME2,
  PRESS 2. ]
SETCURSOR [ 5 11 ] PRINT [ ( IT'S HARDER! ) ]
SETCURSOR [ 0 14 ] PRINT [ TYPE A NUMBER
  AND THEN PRESS RETURN. ]
SETCURSOR [ 16 17 ]
MAKE "C READLIST
IF :C=[ 1 ] [ GAME1 STOP ]
IF :C=[ 2 ] [ GAME2 STOP ]
HELP.MSG
HELP
END


TO HELP.MSG
SETCURSOR [ 0 20 ] PRINT [ PLEASE USE 1 OR
  2 ONLY. ]
SETCURSOR [ 0 22 ] PRINT [ TRY AGAIN —
  PRESS RETURN. ]
MAKE "Q RC
END


TO GAME1
PIC1
SETUP
END

TO GAME2
PIC2
SETUP
END


TO SETUP
PU SETH 0
SETPOS [ 0 -30 ]
ST
PD SETPC 4
MAKE "SCORE 0
PLAY1
PLAY
END
```

```
TO PLAY1
CLEARTEXT
PRINT [ CAN YOU PUT TURTLE IN THE BOX ]
PRINT [ WITHOUT TOUCHING ANY OF THE
  DESIGNS? ]
PRINT [ .....PRESS RETURN TO BEGIN. ]*
MAKE "Q RC
CLEARTEXT
END


TO PLAY
SETCURSOR [ 0 20 ] PRINT [ TYPE IN A
  COMMAND — THEN PRESS RETURN. ]
MAKE "TRY READLIST
IF :TRY=[ ERASE ] [ CHGPEN STOP ]
IF :TRY=[ QUIT ] [ QUIT ]
IF :TRY= [ ] [ BLANK STOP ]
CKINPUT :TRY
SCORES
SETCURSOR [ 0 23 ] TYPE [ ( ..QUIT..OR..
  ERASE ) ]
CHECKBOX
PLAY
END


TO CKINPUT :TRY
TEST COUNT :TRY=2
IFT [ TEST ( OR FIRST :TRY="FD FIRST :TRY=
  "BK FIRST :TRY="RT FIRST :TRY= "LT ) ]
IFT [ TEST NUMBERP FIRST BUTFIRST :TRY ]
IFT [ TEST NUMBERLIMIT FIRST BUTFIRST
  :TRY ]
IFT [ RUN :TRY ]
IFF [ INPUT.MSG ]
END


TO SCORES
CLEARTEXT
MAKE "SCORE :SCORE+1
SETCURSOR [ 29 23 ] TYPE SE [ SCORE: ]
  :SCORE
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 2
APPLE

```
TO CHGPEN
CLEARTEXT
CKINPUT1
SCORES
SETCURSOR [ 0 23 ] TYPE [ ( ..QUIT..OR..
  ERASE ) ]*
CHECKBOX
PD
PLAY
END


TO QUIT
CLEARTEXT
PRINT [ TO PLAY AGAIN...TYPE "HELP". ]*
THROW "TOPLEVEL
END


TO BLANK
CLEARTEXT
BLANK1
CLEARTEXT
PLAY
END


TO NUMBERLIMIT :NUM
IF OR :NUM < 0 :NUM > 360 [ OUTPUT
  LIMIT.MSG ]
OUTPUT "TRUE
END


TO CKINPUT1
CLEARTEXT
SETCURSOR [ 0 20 ] PRINT [ ENTER COMMAND
  FOR ERASING ( PENERASE ) ]
SETCURSOR [ 0 21 ] PRINT [ THEN PRESS
  RETURN. ]
MAKE "E READLIST
IF :E=[] [ BLANK 1 CKINPUT1 STOP ]
TEST OR FIRST :E="PENERASE FIRST :E="PE
IFT [ TEST BF :E=[] ]
IFT [ RUN :E CKINPUT2 ]
IFF [ PE.MSG1 ]
END


TO BLANK1
PRINT [ PLEASE ENTER A COMMAND BEFORE
  YOU ]
PRINT [ PRESS RETURN! ]
PRINT [ .....PRESS RETURN TO TRY AGAIN. ]*
MAKE "Q RC
END

TO INPUT.MSG
SETCURSOR [ 0 23 ] TYPE [ INPUT ERROR —
  TRY AGAIN — PRESS RETURN. ]
MAKE "Q RC
CLEARTEXT
PLAY
END
```

```
TO BOX.MSG
CLEARTEXT
PRINT SE [ GAME OVER!...SCORE: ] :SCORE*
PRINT [ TYPE "HELP" FOR NEW GAME. ]
THROW "TOPLEVEL
END


TO CHECKBOX
TEST AND (XCOR > -129) (XCOR < -90)
IFT [ TEST AND YCOR > 64 YCOR < 104 ]
IFT [ BOX.MSG ]
END


TO CKINPUT2
CLEARTEXT
SETCURSOR [ 0 20 ] PRINT [ ENTER COMMAND
  FOR HOW MUCH TO ERASE. ]
SETCURSOR [ 0 21 ] PRINT [ USE FD OR BK
  WITH A NUMBER.....RETURN. ]*
MAKE "E READLIST
IF :E=[] [ BLANK1 CKINPUT2 STOP ]
TEST COUNT :E=2
IFT [ TEST OR FIRST :E="FD FIRST :E= "BK ]
IFT [ TEST NUMBERP LAST :E ]
IFT [ TEST NUMBERLIMIT LAST :E ]
IFT [ RUN :E ]
IFF [ PE.MSG2 ]
END


TO PE.MSG1
SETCURSOR [ 0 22 ] PRINT [ TO ERASE TYPE
  PENERASE ( OR PE ) ONLY. ]
SETCURSOR [ 26 23 ] TYPE [ PRESS RETURN ]
MAKE "Q RC
CKINPUT1
END


TO PE.MSG2
SETCURSOR [ 0 23 ] TYPE [ OOPS! TRY
  AGAIN — PRESS RETURN ]
MAKE "Q RC
CKINPUT2
END


TO LIMIT.MSG
SETCURSOR [ 0 20 ] PRINT [ NOT A SENSIBLE
  ESTIMATE... ]*
OUTPUT "FALSE
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 3
APPLE

```
TO PIC1
CLEARSCREEN HT
SETBG 0
BORDER
FIELD
PU SETPOS [ -139 30 ]
MTNS 10
STREAM
PU SETPOS [ -60 74 ]
TREE 15
PU SETPOS [ -30 90 ]
TREE 15
FOREST
END

TO PIC2
PIC1
PU SETPOS [ -117 5 ]
TREE 15
PU SETPOS [ -117 -10 ]
TREE 15
PU SETPOS [ 30 50 ]
MTNS 10
PU SETPOS [ 60 80 ]
MTNS 10
PU SETPOS [ 20 70 ]
TREE 15
END

TO BORDER
PU SETPOS [ -139 -76 ]
PD SETPC 1 FD 196
RT 90 FD 278
RT 90 FD 196
RT 90 FD 279
END

TO FIELD
PU SETPOS [ -130 64 ]
SETH 0 SETPC 4
PD SQUARE 40
END

TO MTNS :NUM
SETH 45 PD SETPC 3
REPEAT 4 [ FD :NUM RT 90 FD :NUM LT 90 ]
END

TO STREAM
PU SETPOS [ -95 5 ]
SETH 45 PD SETPC 5
ARCR 5 LT 45 ARCL 5
RT 45
ARCR 5 LT 45 ARCL 5
END

TO TREE :NUM
SETPC 2
PD SETH 0
FD 5 RT 90
BK :NUM / 2
REPEAT 3 [ FD :NUM LT 120 ]
END

TO FOREST
PU SETPOS [ -10 20 ] TREE 10
PU SETPOS [ 0 20 ] TREE 10
PU SETPOS [ 10 20 ] TREE 10
PU SETPOS [ 20 20 ] TREE 10
PU SETPOS [ 30 20 ] TREE 10
PU SETPOS [ 40 20 ] TREE 10
END

TO SQUARE :NUM
REPEAT 4 [ FD :NUM RT 90 ]
END

TO ARCL :NUM
REPEAT 6 [ FD :NUM LT 10 ]
END

TO ARCR :NUM
REPEAT 6 [ FD :NUM RT 10 ]
END
```

Session 8
Procedures for Game, Page 1
ATARI

This game provides practice in estimating distances and degrees and in using the primitive for erasing. The object of the game is to put Turtle in the box in the upper left corner of the screen using as few commands as possible (FD, BK, RT or LT). If a shape is touched, the part of the line touching the obstacle must be erased using the erase command. Each command must be executed before a new command can be entered. Each command is counted as one move in the scorekeeping process.

Enter the following procedures into the computer. Try the game to make sure it runs correctly. Game 2 is a slightly more difficult version of Game 1. Save the procedures under the name GAME8 or something suitable.

You will need to load the game procedures into the computer using the name you saved them under. Then students activate the game by typing HELP.

```
TO HELP
CT TS SETCURSOR [ 0 0 ]
PRINT [ .....ESTIMATING..... ]*
SETCURSOR [ 0 7 ] PRINT [ TO PLAY GAME1,
  PRESS 1. ]
SETCURSOR [ 0 10 ] PRINT [ TO PLAY GAME2,
  PRESS 2. ]
SETCURSOR [ 5 11 ] PRINT [ ( IT'S HARDER! ) ]
SETCURSOR [ 0 14 ] PRINT [ TYPE A NUMBER
  AND THEN PRESS RETURN. ]
SETCURSOR [ 16 17 ]
MAKE "C RL
IF :C=[ 1 ] [ GAME1 STOP ]
IF :C=[ 2 ] [ GAME2 STOP ]
HELP.MSG
HELP
END


TO HELP.MSG
SETCURSOR [ 0 20 ] PRINT [ PLEASE USE 1 OR
  2 ONLY. ]
SETCURSOR [ 0 22 ] PRINT [ TRY AGAIN —
  PRESS RETURN. ]
MAKE "Q RC
END


TO GAME1
PIC1
SETUP
END

TO GAME2
PIC2
SETUP
END


TO SETUP
PU SETH 0
SETPOS [ 0 -30 ]
ST
PD
MAKE "SCORE 0
PLAY1
PLAY
END
```

```
TO PLAY1
CT
PRINT [ CAN YOU PUT TURTLE IN THE BOX ]
PRINT [ WITHOUT TOUCHING ANY OF THE
  DESIGNS? ]
PRINT [ .....PRESS RETURN TO BEGIN. ]*
MAKE "Q RC
CT
END


TO PLAY
SETCURSOR [ 0 20 ] PRINT [ TYPE IN A
  COMMAND — THEN PRESS RETURN. ]
MAKE "TRY RL
IF :TRY=[ ERASE ] [ CHGPEN STOP ]
IF :TRY=[ QUIT ] [ QUIT ]
IF :TRY= [] [ BLANK STOP ]
CKINPUT :TRY
SCORES
SETCURSOR [ 0 23 ] TYPE [ ( ..QUIT..OR..
  ERASE ). ]*
CHECKBOX
PLAY
END


TO CKINPUT :TRY
IF NOT ( ( COUNT :TRY )=2 ) [ INPUT.MSG
  STOP ]
IF OR ( NOT ( MOVESP FIRST :TRY ) ) ( NOT
  ( NUMBERP LAST :TRY ) ) [INPUT.MSG STOP ]
IF NUMBERLIMIT LAST :TRY [ RUN :TRY ]
  [ INPUT.MSG ]
END


TO SCORES
CT
MAKE "SCORE :SCORE+1
SETCURSOR [ 27 23 ] TYPE SE [ SCORE: ]
  :SCORE
END


TO CHECKBOX
IF ( AND ( XCOR > -129 ) ( XCOR < -90 )
  ( YCOR > 64 ) ( YCOR <104 ) ) [ BOX.MSG ]
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 2
ATARI

```
TO CHGPEN
CT
CKINPUT1
SCORES
SETCURSOR [ 0 23 ] TYPE [ ( ..QUIT..OR..
  ERASE ).]
CHECKBOX
PD
PLAY
END


TO QUIT
CT TS
SETCURSOR [ 17 10 ] PRINT [ TO END, ]
SETCURSOR [ 7 12 ] PRINT [ PLEASE PRESS
  THE BREAK KEY. ]
SETCURSOR [ 0 23 ]
MAKE "Q RC
END


TO BLANK
CT
BLANK1
CT
PLAY
END


TO MOVESP :DIR
IF ( OR ( :DIR="FD ) ( :DIR="BK ) ( :DIR= "RT )
  ( :DIR="LT ) ) [ OUTPUT "TRUE ] [ OUTPUT
  FALSE ]
END


TO NUMBERLIMIT :NUM
IF OR ( :NUM < 0 ) ( :NUM > 360 ) [ OUTPUT
  LIMIT.MSG ] [ OUTPUT "TRUE ]
END


TO CKINPUT1
CT
SETCURSOR [ 0 20 ] PRINT [ ENTER COMMAND
FOR ERASING ( PE ).]
SETCURSOR [ 0 21 ] PRINT [ THEN PRESS
  RETURN. ]
MAKE "E RL
IF :E=[] [ BLANK1 CKINPUT1 STOP ]
IF AND ( ( FIRST :E )="PE ) ( ( BF :E=[] )
  [ RUN :E CKINPUT2 ] [ PE.MSG1 ]
END


TO BLANK1
PRINT [ PLEASE ENTER A COMMAND BEFORE
  YOU ]
PRINT [ PRESS RETURN! ]
PRINT [ .....PRESS RETURN TO TRY AGAIN. ]*
MAKE "Q RC
END


TO INPUT.MSG
SETCURSOR [ 0 23 ] TYPE [ ERROR —
  TRY AGAIN — PRESS RETURN / / / ]
MAKE "Q RC
CT
PLAY
END


TO BOX.MSG
CT
PRINT SE [ GAME OVER!...SCORE: ] :SCORE*
PRINT [ TO END, PLEASE PRESS THE BREAK
  KEY. ]
MAKE "Q RC
END


TO CKINPUT2
CT
SETCURSOR [ 0 20 ] PRINT [ ENTER COMMAND
  FOR HOW MUCH TO ERASE. ]
SETCURSOR [ 0 21 ] PRINT [ ( FD OR BK WITH
  A NUMBER. ) ... RETURN ]*
MAKE "E RL
IF :E=[] [BLANK1 CKINPUT2 STOP ]
IF NOT ( ( COUNT :E )=2 ) [ PE.MSG2 STOP ]
IF OR ( NOT (MOVES1P FIRST :E ) ) ( NOT
  ( NUMBERP LAST :E ) ) [ PE.MSG2 STOP ]
IF NUMBERLIMIT LAST :E [ RUN :E ]
  [ PE.MSG2 ]
END


TO PE.MSG1
SETCURSOR [ 0 22 ] PRINT [ TO ERASE, TYPE
  PE. ]
SETCURSOR [ 25 23 ] TYPE [ PRESS RETURN. ]
MAKE "Q RC
CKINPUT1
END


TO PE.MSG2
SETCURSOR [ 0 21 ] TYPE [ OOPS! TRY
  AGAIN — PRESS RETURN. / / / ]
MAKE "Q RC
CKINPUT2
END


TO MOVES1P :DIR
IF OR ( :DIR="FD ) ( :DIR="BK ) [ OUTPUT
  "TRUE ] [ OUTPUT "FALSE ]
END


TO LIMIT.MSG
SETCURSOR [ 0 20 ] PRINT [ NOT A SENSIBLE
  ESTIMATE... ]*
OUTPUT "FALSE
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 3
ATARI

```
TO PIC1
CS HT
SETBG 1
BORDER
FIELD
PU SETPOS [ -139 30 ]
MTNS 10
STREAM
PU SETPOS [ -60 74 ]
TREE 15
PU SETPOS [ -30 90 ]
TREE 15
FOREST
END


TO PIC2
PIC1
PU SETPOS [ -117 5 ]
TREE 15
PU SETPOS [ -117 -10 ]
TREE 15
PU SETPOS [ 30 50 ]
MTNS 10
PU SETPOS [ 60 80 ]
MTNS 10
PU SETPOS [ 20 70 ]
TREE 15
END


TO BORDER
PU SETPOS [ -139 -76 ]
PD
SETPN 0
SETPC 0 26
FD 196
RT 90 FD 278
RT 90 FD 196
RT 90 FD 279
END


TO FIELD
PU SETPOS [ -130 64 ]
SETH 0
SETPN 0
SETPC 0 26
PD SQUARE 40
END
```

```
TO MTNS :NUM
SETH 45 PD
SETPN 1
SETPC 1 90
REPEAT 4 [ FD :NUM RT 90 FD :NUM LT 90 ]
END


TO STREAM
PU SETPOS [ -95 5 ]
SETH 45 PD
SETPN 1
SETPC 1 90
ARCR 5 LT 45 ARCL 5 RT 45
ARCR 5 LT 45 ARCL 5
END


TO TREE :NUM
SETPN 2
SETPC 2 111
PD SETH 0
FD 5 RT 90
BK :NUM / 2
REPEAT 3 [ FD : NUM LT 120 ]
END


TO FOREST
PU SETPOS [ -10 20 ] TREE 10
PU SETPOS [ 0 20 ] TREE 10
PU SETPOS [ 10 20 ] TREE 10
PU SETPOS [ 20 20 ] TREE 10
PU SETPOS [ 30 20 ] TREE 10
PU SETPOS [ 40 20 ] TREE 10
END


TO SQUARE :NUM
REPEAT 4 [ FD :NUM RT 90 ]
END


TO ARCR :NUM
REPEAT 6 [ FD :NUM RT 10 ]
END


TO ARCL :NUM
REPEAT 6 [ FD :NUM LT 10 ]
END
```

Session 8
Procedures for Game, Page 1
MIT (Apple Computer)

This game provides practice in estimating distances and degrees and in using the primitive for erasing. The object of the game is to put Turtle in the box in the upper left corner of the screen using as few commands as possible (FD, BK, RT or LT). If a shape is touched, the part of the line touching the obstacle must be erased using the erase command. Each command must be executed before a new command can be entered. Each command is counted as one move in the scorekeeping process.

The following procedures draw the *gameboard* for Game1. You save the screen image of the gameboard by using the **SAVEPICT** command. During actual game play a procedure calls the gameboard picture from the diskette using the **READPICT** command.

Enter the following procedures into the computer. Type the command **PIC1**. Insert the diskette you will use for storing the game procedures. When the gameboard is complete, type **SAVEPICT "GAME1** and press RETURN.

```
TO PIC1
  BACKGROUND 0
  DRAW
  ORANGEBOX
  BAR
  TRUCKS
  POST
END

TO ORANGEBOX
  PC 4
  PU
  ORANGELINES - 129
END

TO BAR
  PU SETXY - 99 ( - 40 )
  PD SETH 0
  SPIRAL 1
  RECTANGLE1 - 99 ( - 40 )
  PU
  SETXY 29 10
  PD SETH 0
  SPIRAL 1
  RECTANGLE2 - 60 60
  PU SETXY - 35 100
  PD SETH 0
  SPIRAL 1
  END

TO TRUCKS
  PC 5
  RECTANGLE3 - 137 20
  DOT - 137 18
  DOT - 117 18
  DOT - 105 18
  DOT - 99 18
  DOT - 79 18
  DOT - 67 18
END

TO POST
  PU SETXY 139 ( - 125 )
  SETH 0
  PD PC 1
  FD 240
  PU
END

TO ORANGELINES :X
  IF :X = - 90 STOP
  SETH 0
  SETXY :X 64
  PD FD 40
  ORANGELINES :X + 1
END

TO SPIRAL :S
  IF :S = 15 STOP
  PC 3
  FD :S
  RT 60
  SPIRAL :S + .5
END

TO RECTANGLE1 :X :Y
  IF :Y = 10 STOP
  PC 2
  PU SETH 0
  SETXY :X :Y
  PD
  REPEAT 2 [ FD 10 RT 90 FD 40 RT 90 ]
  FD 5 RT 90 FD 40
  RECTANGLE1 :X + 20 :Y + 10
END

TO RECTANGLE2 :X :Y
  IF :Y = 90 STOP
  PC 2
  PU SETH 270
  SETXY :X :Y
  RECTANGLE
  FD 5 RT 90 FD 20
  RECTANGLE2 :X + 10 :Y + 10
END
```

Session 8
Procedures for Game, Page 2
MIT (Apple Computer)

```
TO RECTANGLE3 :X :Y
  IF :X = - 61 STOP
  PU SETH 0
  SETXY :X :Y
  RECTANGLE
  PU LT 90
  BK 32 PD
  CAB
  RECTANGLE3 :X+38 :Y
  END


TO RECTANGLE
  PD
  REPEAT 2 [FD 10 RT 90 FD 20 RT 90 ]
END
```

```
TO DOT :D :E
  PU SETH 90
  SETXY :D :E
  PD FD 1
END


TO CAB
  REPEAT 2 [ FD 8 RT 90 ]
  FD 4 RT 90
  FD 2 LT 90
  FD 4 RT 90
  FD 6
END
```

The following procedures draw the gameboard for Game2, a slightly harder version of Game1. Enter the following procedures into the computer. Type the command **PIC2**. Using the same diskette that you used for **PIC1**, save the gameboard picture by typing **SAVEPICT "GAME2** and pressing RETURN.

```
TO PIC2
  PIC1
  ROAD
  POST2
  WHITEBAR
  TENT - 50 35
  TENT 132 0
  HT
END


TO ROAD
  PU SETXY - 137 16
  SETH 90 PD PC 1
  FD 80
END


TO POST2
  PU SETXY - 130 ( - 70 )
  SETH 0
  PD PC 1
  FD 55
END


TO WHITEBAR
  PC 1
  RECTANGLE4 10 70
END
```

```
TO TENT :X :Y
  PC 4
  PU SETH 270
  SETXY :X :Y
  PD TRI 10
END


TO RECTANGLE4 :X :Y
  IF :Y= - 20 STOP
  PU SETH 0
  SETXY :X :Y
  RECTANGLE
  FD 4 RT 90 FD 20
  RECTANGLE4 :X+10 :Y-10
END


TO TRI :T
  IF :T=18 STOP
  FD :T
  RT 120
  TRI :T+1
END
```

Session 8
Procedures for Game, Page 3
MIT (Apple Computer)

Enter the following procedures into the computer after clearing the workspace. Try the game to make sure it runs correctly. Be sure the disk containing the *gameboards* is in the disk drive. Save the procedures under the name GAME8 or something suitable.

You will need to load the game procedures into the computer using the name you saved them under. Then students activate the game by typing HELP.

```
TO HELP
  NODRAW
  CLEARTEXT
  PRINT [ ..... ESTIMATING ..... ]*
  CURSOR 0 4 PRINT [ TO SEE THE
    DIRECTIONS, PRESS D. ]
  CURSOR 0 7 PRINT [ TO PLAY GAME1, PRESS
    1. ]
  CURSOR 0 10 PRINT [ TO PLAY GAME2,
    PRESS 2. ]
  CURSOR 5 11 PRINT [ (IT'S HARDER!) ]
  CURSOR 0 13 PRINT [ SELECT D, 1 OR 2
    THEN PRESS RETURN. ]
  CURSOR 17 16
  MAKE "C REQUEST
  IF :C=[ D ] DIR1 STOP
  IF :C=[ 1 ] GAME1 STOP
  IF :C=[ 2 ] GAME2 STOP
  ER.MSG
  HELP
END

TO DIR1
  CLEARTEXT
  CURSOR 0 1 PRINT [ YOUR TASK IS TO PUT
    TURTLE IN THE BOX ]
  CURSOR 0 3 PRINT [ IN THE UPPER LEFT
    CORNER WITHOUT ]
  CURSOR 0 5 PRINT [ TOUCHING ANY OF THE
    DESIGNS. ]
  CURSOR 0 7 PRINT [ USE THE COMMANDS FD,
    BK, RT OR LT WITH ]
  CURSOR 0 9 PRINT [ ANY NUMBER YOU
    CHOOSE. ]
  CURSOR 3 20 PRINT [ PRESS RETURN. ]
  MAKE "Q RC
  DIR2
END

TO DIR2
  CLEARTEXT
  CURSOR 0 2
  PRINT [ LOGO WILL KEEP SCORE BY
    COUNTING ]
  CURSOR 0 4 PRINT [ HOW MANY COMMANDS
    YOU GIVE THE ]
  CURSOR 0 6 PRINT [ TURTLE TO GET IT IN
    THE BOX. ]
  CURSOR 0 8 PRINT [ THE WINNER IS THE
    PERSON WHO USES THE ]
  CURSOR 0 10 PRINT [ FEWEST COMMANDS. ]
  CURSOR 0 12 PRINT [ ONLY ONE COMMAND
    AT A TIME! ]
  CURSOR 3 20 PRINT [ PRESS RETURN. ]
  MAKE "Q RC
  DIR3
END

TO DIR3
  CLEARTEXT
  CURSOR 0 1 PRINT [ READY TO BEGIN? ]
  CURSOR 3 3 PRINT [ PRESS 1 FOR GAME1. ]
  CURSOR 3 5 PRINT [ PRESS 2 FOR GAME2. ]
  CURSOR 3 7 PRINT [ PRESS D FOR
    DIRECTIONS. ]
  CURSOR 3 10 PRINT [ ..... THEN PRESS
    RETURN. ]*
  CURSOR 14 13
  MAKE "C REQUEST
  IF :C=[ 1 ] GAME1 STOP
  IF :C=[ 2 ] GAME2 STOP
  IF :C=[ D ] DIR1 STOP
  ER.MSG
  DIR3
END

TO GAME1
  READPICT "GAME1
  SETUP
END

TO GAME2
  READPICT "GAME2
  SETUP
END

TO SETUP
  PU SETH 0
  SETXY 0 ( - 30 )
  PD PC 4
  MAKE "SCORE 0
  PLAY1
  PLAY
END

TO PLAY1
  CLEARTEXT
  PRINT [ CAN YOU PUT TURTLE IN THE
    BOX ]
  PRINT [ WITHOUT TOUCHING ANY OF THE
    DESIGNS? ]
  PRINT [ ..... PRESS RETURN TO BEGIN. ]*
  MAKE "Q RC
  CLEARTEXT
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 4
MIT (Apple Computer)

```
TO PLAY
  CURSOR 0 20 PRINT [ TYPE IN A COMMAND
    — THEN PRESS RETURN. ]
  MAKE "TRY REQUEST
  IF :TRY=[ ERASE ] CHGPEN STOP
  IF :TRY=[ QUIT ] QUIT
  IF :TRY=[ ] BLANK STOP
  CKINPUT :TRY
  SCORES
  CURSOR 0 23 PRINT1 [ ( .. QUIT .. OR ..
    ERASE. ) ]
  CHECKBOX
  PLAY
END

TO CKINPUT :TRY
  TEST COUNTS :TRY=2
  IFT TEST ( ANYOF ( FIRST :TRY="FD )
    ( FIRST :TRY="BK ) ( FIRST :TRY= "RT )
    ( FIRST :TRY="LT ) )
  IFT TEST NUMBER? LAST :TRY
  IFT TEST NUMBERLIMIT LAST :TRY
  IFT RUN :TRY STOP
  IFF INPUT.MSG
END


TO CHGPEN
  CLEARTEXT
  CKINPUT1
  SCORES
  CURSOR 0 23 PRINT1 [ ( .. QUIT .. OR ..
    ERASE ) ]*
  CHECKBOX
  PC 4
  PLAY
END


TO CKINPUT1
  CLEARTEXT
  CURSOR 0 20 PRINT [ ENTER COMMAND FOR
    ERASING ( PENCOLOR 0 ). ]
  CURSOR 0 21 PRINT [ ..... PRESS RETURN. ]*
  MAKE "E REQUEST
  IF :E=[ ] BLANK1 CKINPUT1 STOP
  TEST COUNTS :E=2
  IFT TEST ANYOF ( FIRST :E="PENCOLOR )
    ( FIRST :E="PC )
  IFT TEST LAST :E=0
  IFT RUN :E CKINPUT2 STOP
  IFF PC.MSG1
END


TO CKINPUT2
  CLEARTEXT
  CURSOR 0 20
  PRINT [ ENTER COMMAND FOR HOW MUCH
    TO ERASE. ]
  CURSOR 0 21
  PRINT [ USE FD OR BK WITH A NUMBER.—
    RETURN. ]
  MAKE "E REQUEST
  IF :E=[ ] BLANK1 CKINPUT2 STOP
  TEST COUNTS :E=2
  IFT TEST ANYOF ( FIRST :E="FD ) ( FIRST
    :E="BK )
  IFT TEST NUMBER? LAST :E
  IFT TEST NUMBERLIMIT LAST :E
  IFT RUN :E STOP
  IFF PC.MSG2
END


TO SCORES
  CLEARTEXT
  MAKE "SCORE :SCORE+1
  CURSOR 29 23 PRINT1 SE [ SCORE: ] :SCORE
END


TO CHECKBOX
  TEST ALLOF XCOR > - 129 XCOR < - 90
  IFT TEST ALLOF YCOR > 64 YCOR < 104
  IFT BOX.MSG
END


TO COUNTS :LISTS
  IF :LISTS=[ ] OUTPUT 0
  OUTPUT 1+COUNTS BUTFIRST :LISTS
END


TO NUMBERLIMIT :NUM
  IF ANYOF ( :NUM < 0 ) ( :NUM > 360 )
    OUTPUT LIMIT.MSG
  OUTPUT "TRUE
END


TO ER.MSG
  CURSOR 0 19 PRINT [ PLEASE USE D, 1 OR
    2. ]
  CURSOR 5 20 PRINT [ PRESS RETURN TO
    RE-ENTER. ]
  MAKE "Q RC
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 5
MIT (Apple Computer)

```
TO BLANK
  CLEARTEXT BLANK1
  CLEARTEXT PLAY
END


TO BLANK1
  PRINT [ ENTER A COMMAND BEFORE
    PRESSING RETURN. ]
  PRINT [ ..... PRESS RETURN TO TRY AGAIN. ]*
  MAKE "Q RC
END


TO INPUT.MSG
  CURSOR 0 23 PRINT1 [INPUT ERROR — TRY
    AGAIN — PRESS RETURN. ]
  MAKE "Q RC
  CLEARTEXT
  PLAY
END


TO PC.MSG1
  CURSOR 0 22 PRINT [ TO ERASE TYPE
    PENCOLOR 0 ( OR PC 0 ). ]
  CURSOR 26 23 PRINT1 [ PRESS RETURN. ]
  MAKE "Q RC
  CKINPUT1
END
```

```
TO PC.MSG2
  CURSOR 0 23 PRINT1 [ OOPS! ..... PRESS
    RETURN AND TRY AGAIN. ]*
  MAKE "Q RC
  CKINPUT2
END


TO BOX.MSG
  CLEARTEXT
  PRINT SE [ GAME OVER! ... SCORE: ] :SCORE*
  PRINT [ TYPE: HELP, FOR NEW GAME. ]
  TOPLEVEL
END


TO LIMIT.MSG
  CURSOR 0 20 PRINT [ NOT A SENSIBLE
    ESTIMATE ... ]*
  OUTPUT "FALSE
END


TO QUIT
  CLEARTEXT
  PRINT [ TO PLAY AGAIN ... TYPE: HELP. ]*
  TOPLEVEL
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 1
MIT (Commodore 64)

This game provides practice in estimating distances and degrees and in using the primitive for erasing. The object of the game is to put Turtle in the box in the upper left corner of the screen using as few commands as possible (FD, BK, RT or LT). If a shape is touched, the part of the line touching the obstacle must be erased using the erase command. Each command must be executed before a new command can be entered. Each command is counted as one move in the scorekeeping process.

The following procedures draw the *gameboard* for Game1. You save the screen image of the gameboard by using the **SAVEPICT** command. During actual game play a procedure calls the gameboard picture from the diskette using the **READPICT** command.

Enter the following procedures into the computer. Type the command **PIC1**. Insert the diskette you will use for storing the game procedures. When the gameboard is complete, type **SAVEPICT "GAME1** and press RETURN.

```
TO PIC1
  BACKGROUND 0
  DRAW
  ORANGEBOX
  BAR
  TRUCKS
  POST
END


TO ORANGEBOX
  PC 8
  PU
  ORANGELINES - 159
END


TO BAR
  PU SETXY - 99 ( - 40 )
  PD SETH 0
  SPIRAL 1
  RECTANGLE1 - 99 ( - 40 )
  PU
  SETXY 29 10
  PD SETH 0
  SPIRAL 1
  RECTANGLE2 - 60 60
  PU SETXY - 35 100
  PD SETH 0
  SPIRAL 1
  END

TO TRUCKS
  PC 5
  RECTANGLE3 - 137 20
  DOT - 137 18
  DOT - 117 18
  DOT - 105 18
  DOT - 99 18
  DOT - 79 18
  DOT - 67 18
END
```

```
TO POST
  PU
  SETXY 145 ( - 125 )
  SETH 0
  PD PC 1
  FD 250
  SETXY 147 ( - 125 )
  SETH 0
  FD 250
  PU
END


TO ORANGELINES :X
  IF :X = - 120 STOP
  SETH 0
  SETXY :X 64
  PD FD 40
  ORANGELINES :X+1
END


TO SPIRAL :S
  IF :S=15 STOP
  PC 13
  FD :S
  RT 60
  SPIRAL :S+0.5
END


TO RECTANGLE1 :X :Y
  IF :Y=10 STOP
  PC 7
  PU SETH 0
  SETXY :X :Y
  PD
  REPEAT 2 [ FD 10 RT 90 FD 40 RT 90 ]
  FD 5 RT 90 FD 40
  RECTANGLE1 :X+20 :Y+10
END
```

Session 8
Procedures for Game, Page 2
MIT (Commodore 64)

```
TO RECTANGLE2 :X :Y
  IF :Y=90 STOP
  PC 5
  PU SETH 270
  SETXY :X :Y
  RECTANGLE
  FD 5 RT 90 FD 20
  RECTANGLE2 :X+10 :Y+10
END

TO RECTANGLE3 :X :Y
  IF :X = - 61 STOP
  PU SETH 0
  SETXY :X :Y
  RECTANGLE
  PU LT 90
  BK 32 PD
  CAB
  RECTANGLE3 :X+38 :Y
END
```

```
TO RECTANGLE
  PD
  REPEAT 2 [ FD 10 RT 90 FD 20 RT 90 ]
END

TO DOT :D :E
  PU SETH 90
  SETXY :D :E
  PD FD 1
END

TO CAB
  REPEAT 2 [ FD 8 RT 90 ]
  FD 4 RT 90
  FD 2 LT 90
  FD 4 RT 90
  FD 6
END
```

The following procedures draw the gameboard for Game2, a slightly harder version of Game1. Enter the following procedures into the computer. Type the command **PIC2**. Using the same diskette that you used for **PIC1**, save the gameboard picture by typing **SAVEPICT "GAME2** and pressing RETURN.

```
TO PIC2
  PIC1
  ROAD
  POST2
  WHITEBAR
  TENT - 50 35
  TENT 132 0
  HT
END

TO ROAD
  PU SETXY - 160 16
  SETH 90 PD PC 7
  FD 100
END

TO POST2
  PU SETXY - 150 ( - 70 )
  SETH 0
  PD PC 1
  FD 55
END

TO WHITEBAR
  PC 1
  RECTANGLE4 10 70
END
```

```
TO TENT :X :Y
  PC 4
  PU SETH 270
  SETXY :X :Y
  PD TRI 10
END

TO RECTANGLE4 :X :Y
  IF :Y= - 20 STOP
  PU SETH 0
  SETXY :X :Y
  RECTANGLE
  FD 5 RT 90 FD 20
  RECTANGLE4 :X+10 :Y-10
END

TO TRI :T
  IF :T=18 STOP
  FD :T
  RT 120
  TRI :T+1
END
```

Session 8
Procedures for Game, Page 3
MIT (Commodore 64)

Enter the following procedures into the computer after clearing the workspace. Try the game to make sure it runs correctly. Be sure the disk containing the *gameboards* is in the disk drive. Save the procedures under the name GAME8 or something suitable.

You will need to load the game procedures into the computer using the name you saved them under. Then students activate the game by typing HELP.

```
TO HELP
  NODRAW
  CLEARTEXT
  PRINT [ ..... ESTIMATING ..... ]*
  CURSOR 0 4 PRINT [ TO SEE THE
    DIRECTIONS, PRESS D. ]
  CURSOR 0 7 PRINT [ TO PLAY GAME1, PRESS
    1. ]
  CURSOR 0 9 PRINT [ TO PLAY GAME2,
    PRESS 2. ]
  CURSOR 5 11 PRINT [ (IT'S HARDER!) ]
  CURSOR 0 13 PRINT [ SELECT D, 1 OR 2,
    THEN PRESS RETURN. ]
  CURSOR 17 16
  MAKE "C REQUEST
  IF :C=[ D ] DIR0 STOP
  IF :C=[ 1 ] GAME1 STOP
  IF :C=[ 2 ] GAME2 STOP
  ER.MSG
  HELP.MSG
END


TO DIR0
  READPICT "GAME1
  SPLITSCREEN
  CLEARTEXT
  PRINT [ THIS IS THE GAMEBOARD. ]
  PRINT [ ]
  PRINT [ PRESS RETURN TO SEE THE
    DIRECTIONS. ]
  MAKE "Q RC
  DIR1
END


TO DIR1
  TEXTSCREEN CLEARTEXT
  CURSOR 0 1 PRINT [ YOUR TASK IS TO PUT
    TURTLE IN THE BOX ]
  CURSOR 0 3 PRINT [ IN THE UPPER LEFT
    CORNER WITHOUT ]
  CURSOR 0 5 PRINT [ TOUCHING ANY OF THE
    DESIGNS. ]
  CURSOR 0 7 PRINT [ USE THE COMMANDS
    FD, BK, RT OR LT ]
  CURSOR 0 9 PRINT [ WITH ANY NUMBER YOU
    CHOOSE. ]
  CURSOR 3 20 PRINT [ PRESS RETURN. ]
  MAKE "Q RC
  DIR2
END


TO DIR2
  CLEARTEXT
  CURSOR 0 2 PRINT [ IF YOU GIVE A
    COMMAND THAT CAUSES ]
  CURSOR 0 4 PRINT [ TURTLE'S LINE TO
    TOUCH A DESIGN, YOU ]
  CURSOR 0 6 PRINT [ MUST ERASE THAT
    PART OF TURTLE'S LINE. ]
  CURSOR 0 8 PRINT [ TYPE "PENERASE" AND
    PRESS RETURN. THEN ]
  CURSOR 0 10 PRINT [ YOU MUST ENTER THE
    COMMAND FOR HOW MUCH ]
  CURSOR 0 12 PRINT [ OF THE LINE TO
    ERASE. ( EXAMPLE: BK 30 ). ]
  CURSOR 6 20 PRINT [ PRESS RETURN. ]
  MAKE "Q RC
  DIR3
END


TO DIR3
  CLEARTEXT
  CURSOR 0 2 PRINT [ LOGO WILL KEEP
    SCORE BY COUNTING ]
  CURSOR 0 4 PRINT [ HOW MANY
    COMMANDS YOU GIVE THE ]
  CURSOR 0 6 PRINT [ TURTLE TO GET
    IT IN THE BOX. ]
  CURSOR 0 8 PRINT [ THE WINNER IS THE
    PERSON WHO USES THE ]
  CURSOR 0 10 PRINT [ FEWEST COMMANDS. ]
  CURSOR 0 12 PRINT [ ONLY ONE COMMAND
    AT A TIME! ]
  CURSOR 3 20 PRINT [ PRESS RETURN. ]
  MAKE "Q RC
  DIR4
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 4
MIT (Commodore 64)

```
TO DIR4
  CLEARTEXT
  CURSOR 0 2 PRINT [ TO SEE DIRECTIONS
    AGAIN, TYPE: H (HELP). ]
  CURSOR 0 4 PRINT [ TO QUIT PLAYING, TYPE:
    Q. ]
  CURSOR 0 10 PRINT [ READY TO BEGIN? ]
  CURSOR 3 12 PRINT [ PRESS 1 FOR GAME1. ]
  CURSOR 3 14 PRINT [ PRESS 2 FOR GAME2. ]
  CURSOR 3 17 PRINT [ TO RETURN TO YOUR
    GAME, PRESS R. ]
  CURSOR 19 18
  MAKE "C RC
  CURSOR 0 20 ( PRINT [ YOU PRESSED ]
    :C [ . ] )
  IF :C = "R SPLITSCREEN PLAY STOP
  IF :C = "1 SETUP STOP
  IF :C = "2 GAME2 STOP
  IF :C = "Q QUIT
  IF :C = "H HELP2 STOP
  DIR.MSG
  DIR4
END


TO GAME1
  READPICT "GAME1
  SETUP
END


TO GAME2
  READPICT "GAME2
  SETUP
END


TO HELP2
  SPLITSCREEN CLEARTEXT
  PRINT [ PRESS RETURN FOR DIRECTIONS. ]
  MAKE "Q RC
  DIR1
END


TO SETUP
  SPLITSCREEN
  PU SETH 0
  SETXY 0 ( - 30 )
  PD PC 1
  MAKE "SCORE 0
  PLAY1
  PLAY
END
```

```
TO PLAY1
  CLEARTEXT
  PRINT [ CAN YOU PUT TURTLE IN THE
    BOX ]
  PRINT [ WITHOUT TOUCHING ANY OF THE
    DESIGNS? ]
  PRINT [ ..... PRESS RETURN TO BEGIN. ]*
  MAKE "Q RC
  CLEARTEXT
END


TO PLAY
  CURSOR 0 20
  PRINT [ TYPE IN A COMMAND — THEN
    PRESS RETURN. ]
  MAKE "TRY REQUEST
  IF :TRY = [ HELP ] DIR1 STOP
  IF :TRY = [ PENERASE ] ERASES STOP
  IF :TRY = [ QUIT ] QUIT
  IF :TRY = [ ] BLANK STOP
  CKINPUT :TRY
  SCORES
  CURSOR 0 23 PRINT1 [ ( .. OR .. PENERASE
    .. HELP .. QUIT ) ]*
  CKBOX
  PLAY
END

TO CKINPUT :TRY
  TEST COUNTS :TRY = 2
  IFT TEST ( ANYOF ( FIRST :TRY = "FD )
    ( FIRST :TRY = "BK ) ( FIRST :TRY = "RT )
    ( FIRST :TRY = "LT ) )
  IFT TEST NUMBER? LAST :TRY
  IFT TEST NUMBERLIMIT LAST :TRY
  IFT RUN :TRY STOP
  IFF INPUT.MSG
END


TO ERASES
  RUN :TRY
  CKINPUT1
  SCORES
  CURSOR 0 23 PRINT1 [ ( .. QUIT .. OR ..
    PENERASE ) ]*
  CKBOX
  PC 1
  PLAY
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 5
MIT (Commodore 64)

```
TO CKINPUT1
  CLEARTEXT
  CURSOR 0 20 PRINT [ ENTER COMMAND FOR
    HOW MUCH TO ERASE. ]
  CURSOR 0 21 PRINT [ USE FD OR BK WITH A
    NUMBER — RETURN. ]
  MAKE "E REQUEST
  IF :E=[ ] BLANK1 CKINPUT1 STOP
  TEST COUNTS :E=2
  IFT TEST ANYOF ( FIRST :E="FD )
    ( FIRST :E="BK )
  IFT TEST NUMBER? LAST :E
  IFT TEST NUMBERLIMIT LAST :E
  IFT RUN :E STOP
  IFF ERASE.MSG
END


TO SCORES
  CLEARTEXT
  MAKE "SCORE :SCORE+1
  CURSOR 29 23 PRINT1 SE [ SCORE: ] :SCORE
END


TO CKBOX
  TEST ALLOF XCOR > -  159 XCOR < -  120
  IFT TEST ALLOF YCOR > 64 YCOR < 104
  IFT BOX.MSG
END


TO COUNTS :LISTS
  IF :LISTS=[ ] OUTPUT 0
  OUTPUT 1+COUNTS BUTFIRST :LISTS
END


TO NUMBERLIMIT :NUM
  IF ANYOF ( :NUM < 0 ) ( :NUM > 360 )
    OUTPUT LIMIT.MSG
  OUTPUT "TRUE
END


TO HELP.MSG
  CURSOR 0 19 PRINT [ PLEASE USE D, 1 OR
    2. ]
  CURSOR 5 21 PRINT [ PRESS RETURN TO
    RE-DO. ]
  MAKE "Q RC
END


TO DIR.MSG
  CURSOR 5 21 PRINT [ PLEASE USE 1, 2, R,
    Q OR H. ]
  CURSOR 5 22 PRINT [ PRESS RETURN TO
    TRY AGAIN. ]
  MAKE "Q RC
END
```

```
TO BLANK
  CLEARTEXT
  BLANK1
  CLEARTEXT
  PLAY
END


TO BLANK1
  PRINT [ ENTER A COMMAND BEFORE
    PRESSING RETURN. ]
  PRINT [ ..... PRESS RETURN TO RE-ENTER. ]*
  MAKE "Q RC
END


TO INPUT.MSG
  CURSOR 0 23 PRINT1 [ INPUT ERROR — TRY
    AGAIN — PRESS RETURN. ]
  MAKE "Q RC
  CLEARTEXT
  PLAY
END


TO ERASE.MSG
  CURSOR 0 23 PRINT1 [ OOPS! ..... PRESS
    RETURN AND TRY AGAIN. ]*
  MAKE "Q RC
  CKINPUT1
END


TO BOX.MSG
  CLEARTEXT
  PRINT SE [ GAME OVER! ... SCORE: ] :SCORE*
  PRINT [ TYPE: HELP, FOR NEW GAME. ]
  TOPLEVEL
END


TO LIMIT.MSG
  CURSOR 0 20 PRINT [ NOT A SENSIBLE
    ESTIMATE ... ]*
  OUTPUT "FALSE
END


TO QUIT
  CLEARTEXT
  PRINT [ TO PLAY AGAIN ... TYPE: HELP. ]*
  TOPLEVEL
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 1
TI

This game provides practice in estimating distances and degrees and in using the primitive for erasing. The object of the game is to put Turtle in the box in the upper left corner of the screen using as few commands as possible (FD, BK, RT or LT). If a shape is touched, the part of the line touching the obstacle must be erased using the erase command. Each command must be executed before a new command can be entered. Each command is counted as one move in the scorekeeping process.

Enter the following procedures into the computer. Try the game to make sure it runs correctly. Game 2 is a slightly more difficult version of Game 1. Save the procedures under the name GAME8 or something suitable.

You will need to load the game procedures into the computer using the name you saved them under. Then students activate the game by typing HELP. When a student has directed Turtle into the box, GAME IS OVER will appear on the screen. The student must press FCTN/BACK to stop the procedure. HELP will start the game for the next student.

```
TO HELP
  NOTURTLE
  MAKE "SCORE 0
  MENU
  MAKE "Q RC
  IF :Q="1 GAME1 STOP
  IF :Q="2 GAME2 STOP
  RE.ENTER
END


TO MENU
  CS CLEAR
  PRINT [ .. TO START THE GAME ]*
  PRINT []
  PRINT [ .. PRESS "1" FOR GAME1 ]
  PRINT []
  PRINT [ .. PRESS "2" FOR GAME2 ]
  REPEAT 5 [ PRINT [] ]
  PRINT [ ... PRESS FCTN/BACK WHEN YOU ]
  PRINT [ ... ARE FINISHED WITH A GAME. ]
END


TO RE.ENTER
  REPEAT 5 [ PRINT [] ]
  PRINT [ PLEASE USE 1 OR 2 ]
  PRINT [ ... PRESS ENTER — TRY AGAIN ]*
  MAKE "Q RC
  HELP
END

TO GAME1
  TELL TURTLE
  SC 1 HOME
  CS CLEAR
  REDBOX
  WHITEBAR
  TRUCKS
  TELL TURTLE
  PU HOME
  BK 20 PD
  SC 1
  PLAY
END
```

```
TO GAME2
  TELL TURTLE
  SC 1 HOME
  CS CLEAR
  REDBOX
  WHITEBAR
  TRUCKS
  BLACKBAR
  ROCKETS
  TELL TURTLE HOME
  PU BK 30
  PD SC 1
  PLAY
END


TO REDBOX
  TELL [ 1 2 3 4 ]
  CARRY :BOX
  SS 0 SH 0
  SC :RED
  SXY -80 60
  TELL [ 1 2 ]
  FD 16
  TELL [ 1 3 ]
  SH 270
  FD 16
END


TO TRUCKS
  TELL [ 9 10 ]
  CARRY 2
  SS 0 SH 90
  SC :YELLOW
  SXY -115 20
  TELL 10
  FD 18
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 2
TI

```
TO WHITEBAR
  TELL [ 5 6 7 8 ]
  CARRY :BALL
  SC :BLUE
  SS 0 SH 45
  SXY -50 50
  TELL 5
  FD 42
  TELL 6
  SXY -90 ( -10 )
  TELL 7
  SXY 30 20
  ZIGZAG
END


TO ZIGZAG
  TELL TURTLE
  SC [ 15 15 ]
  SXY 38 20
  SH 253
  FD 120
  SXY -40 42
  SH 45
  FD 35
  SC 1
  HOME
END


TO ROCKETS
  TELL [ 11 12 13 ]
  CARRY 3
  SC 13 SS 0 SH 90
  SXY -80 30
  TELL 11 FD 15
  TELL 12 FD 30
END


TO BLACKBAR
  TELL TURTLE
  SC [ 1 1 ]
  SXY 10 70
  SH 125
  FD 60
  SH 90
  FD 70
END


TO PLAY
  PRINT SE [ SCORE: ] :SCORE
  PRINT []
  PRINT [ TYPE IN COMMAND — PRESS
    ENTER.]
  MAKE "TRY READLINE
  IF :TRY=[] BLANK STOP
  IF :TRY=[ PENERASE ] ER :TRY STOP
  IF :TRY=[ PE ] ER :TRY STOP
  CKINPUT :TRY
  SCORE
  PLAY
END
```

```
TO BLANK
  PRINT [] PRINT []
  PRINT [ PLEASE ENTER A COMMAND ]
  PRINT [ BEFORE YOU PRESS ENTER. ]
  WAIT 125
  REPEAT 4 [ PRINT [] ]
  PLAY
END


TO ER :TRY
  RUN :TRY
  REPEAT 7 [ PRINT [] ]
  PRINT SE [ SCORE: ] :SCORE
  PRINT []
  PRINT [ TURTLE HAS ERASER DOWN. ENTER ]
  PRINT [ FD OR BK WITH A NUMBER. ]
  MAKE "TRY READLINE
  CKINPUT :TRY
  SCORE
  PD PLAY
END


TO CKINPUT :TRY
  TEST EITHER FIRST :TRY="FD FIRST :TRY=
    "BK
  IFF TEST EITHER FIRST :TRY="RT FIRST
    :TRY="LT
  IFT TEST NUMBER? FIRST BF :TRY
  IFT RUN SE FIRST :TRY FIRST BF :TRY
  REPEAT 3 [ PRINT [] ]
  IFF ER.MSG
END


TO CKBOX
  TEST BOTH XCOR > -96 XCOR < -64
  IFT TEST BOTH YCOR < 76 YCOR > 44
  IFT FINAL
END


TO SCORE
  REPEAT 5 [ PRINT [] ]
  MAKE "SCORE :SCORE+1
  CKBOX
END


TO ER.MSG
  REPEAT 5 [ PRINT [] ]
  PRINT [ CAN'T RUN THAT — TRY AGAIN. ]
  WAIT 110
END
```

Session 8
Procedures for Game, Page 3
TI

```
TO FINAL
  PRINT SE [ GAME OVER !! SCORE: ] :SCORE
  PRINT [ PRESS FCTN/BACK ]
  PRINT [ THEN TYPE HELP FOR A NEW
    GAME. ]
  MAKE "Q RC
  ER.MSG
  REPEAT 2 [ PRINT [] ]
  CKBOX
END

TO CLEAR
  TELL :ALL
  CARRY 0
  SS 0 SH 0 SC 0
  SXY -90 ( -80 )
END
```

```
TO QUIT
  CS
  CLEAR
  PRINT [ GAME ENDED. TYPE HELP ... ]*
  PRINT []
  PRINT [ AND PRESS ENTER FOR A ]
  PRINT []
  PRINT [ NEW GAME. ]
END
```

*Use exactly the same number of periods as listed.

Session 8
Procedures for Game, Page 1
TRS

This game provides practice in estimating distances and degrees and in using the primitive for erasing. The object of the game is to put Turtle in the box in the upper left corner of the screen using as few commands as possible (FD, BK, RT or LT). If a shape is touched, the part of the line touching the obstacle must be erased using the erase command. Each command must be executed before a new command can be entered. Each command is counted as one move in the scorekeeping process.

Enter the following procedures into the computer. Try the game to make sure it runs correctly. Save the procedures under the name GAME8 or something suitable.

You will need to load the game procedures into the computer using the name you saved them under. Then students activate the game by typing GAME1 or GAME2. Game2 is a slightly more difficult version of Game1. Students may keep a running score of inputs by typing SCORE after each command is entered. Pressing the BREAK key stops the game. Students may type HELP for instructions. (There is no error trapping in this game because TRS Logo does not include list processing.)

```
TO HELP
HT
CLEAR
LINE 190
PRINT " TO START THE GAME "
LINE 182
PRINT " TYPE GAME1 OR GAME2 "
LINE 174
PRINT " "
LINE 166
PRINT " TO STOP THE GAME "
LINE 158
PRINT " PRESS THE BREAK KEY. "
LINE 150
PRINT " GOOD LUCK "
END


TO LINE :Y
SX 0 SY :Y
END


TO GAME1
CLEAR
COLORSET 1
ORANGEBOX
BAR
TRUCKS
SETUP
END


TO ORANGEBOX
PC 2
ORANGELINES 10
END


TO ORANGELINES :X
IF :X=55 ( STOP )
SETH 0
SETX :X  SETY 143
FD 40
ORANGELINES ( :X+1 )
END
```

```
TO BAR
SX 35 SY 58
SETHEADING 0
SPIRAL 1
RECTANGLE1 40 60
SX 180 SY 125
SETHEADING 0
SPIRAL 1
PECTANGLE2 90 138
SX 105 SY 178
SPIRAL 1
END


TO SPIRAL :S
IF :S=13 ( STOP )
PC 2
FD :S
RT 60
SPIRAL ( :S+1 )
END


TO RECTANGLE1 :X :Y
IF :Y=120 ( STOP )
PC 1
SETHEADING 0
SX :X SY :Y
REPEAT 2 ( FD 10 RT 90 FD 40
 RT 90 )
FD 5 RT 90 FD 40
RECTANGLE1 ( :X+20 ) ( :Y+10 )
END


TO RECTANGLE2 :X :Y
IF :Y=168 ( STOP )
PC 1
SETH 270
SX :X SY :Y
RECTANGLE
FD 5 RT 90 FD 20
RECTANGLE2 ( :X+10 ) (:Y+10 )
END


TO RECTANGLE
REPEAT 2 ( FD 10 RT 90 FD 20
 RT 90 )
END
```

Session 8
Procedures for Game, Page 2
TRS

```
TO TRUCKS
PC 2
RECTANGLE3 2 115
DOT 2 113
DOT 22 113
DOT 36 113
DOT 42 113
DOT 61 113
DOT 74 113
HT
END

TO DOT :D :E
SETH 90
SETX :D SETY :E
FD 1
END

TO RECTANGLE3 :X :Y
IF :X=78 ( STOP )
SETH 0
SETX :X SETY :Y
RECTANGLE
LT 90
BK 32
CAB
RECTANGLE3 ( :X+38 ) :Y
END

TO CAB
REPEAT 2 ( FD 8 RT 90 )
FD 4 RT 90
FD 2 LT 90
FD 4 RT 90
FD 6
END

TO SETUP
PC 0 SETH 0
PLAY1
SETX 128 SETY 65
ST
MAKE :S 0
END

TO PLAY1
LINE 26
PRINT " CAN YOU PUT TURTLE IN "
LINE 18
PRINT " THE BOX WITHOUT "
LINE 10
PRINT " TOUCHING THE DESIGNS? "
SLOW 127  SLOW 127
LINE 26 CLEARTEXT
LINE 18 CLEARTEXT
LINE 10 CLEARTEXT
END

TO CLEARTEXT
PRINT "                      "*
END

TO GAME2
CLEAR
COLORSET 1
ORANGEBOX
BAR
TRUCKS
BLACKBAR
TENT 100 120
TENT 252 84
SETUP
END

TO BLACKBAR
PC 0
RECTANGLE4 180 160
END

TO RECTANGLE4 :X :Y
IF :Y=100 ( STOP )
SETH 0
SETX :X SETY :Y
RECTANGLE
FD 5 RT 90 FD 20
RECTANGLE4 ( :X+10 ) ( :Y-10 )
END

TO TENT :X :Y
PC 0 SETH 270
SETX :X SETY :Y
TRI 10
END

TO TRI :T
IF :T=18 ( STOP )
FD :T
RT 120
TRI ( :T+1 )
END

TO SCORE
HT
MAKE :S :S+1
MAKE :X XLOC ME
MAKE :Y YLOC ME
SX 90 SY 50
PRINT "SCORE= "
SX 155
PRINT :S
SX :X SY :Y
ST
END
```

*Use 22 spaces between the quotes.

Session 9

*"In the Turtle Mathland anthropomorphic images facilitate the transfer of knowledge from familiar settings to new contexts. For example, the metaphor for what is usually called 'programming computers' is teaching the Turtle a new word."**

Seymour Papert
Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVES: To introduce the concept of the **teaching** or **editing** mode and to introduce students to writing procedures.

CONCEPTS/SKILLS: Writing procedures.

FACILITIES/MATERIALS: Computer with Logo cartridge, disk or module; student copies of Keyboard sheet; Key Functions for the Edit Mode wall chart (Charts section); a file box with 4x6 cards preprinted with the procedures TO CRAWL, TO SKIP, TO BAKE CAKE and TO FIX BIKE TIRE; 4x6 cards for each student preprinted with the procedures TO PALC, TO RITONE and TO WELCOME (one procedure per card); and student copies of the Problem Sheet.

PREPARATION: A master for the procedure cards is provided following the ASSIGNMENT section; it can be duplicated on tagboard or photocopied and mounted on tagboard and then cut into 4x6 cards. You may want tagboard or construction paper color to match the edit mode screen color. These cards should be saved as they will be used again.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)

Defining and Editing Procedures, Managing Workspace

TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	

COMMENTS:

Session 9 introduces students to writing, editing and using procedures. This is a lot of material for students to learn. Just to get started, students need to learn:

1. How to enter the edit (procedure definition) mode.
2. Details of a procedure definition (TO line, body, END line).
3. Editing a procedure.
4. Getting back into immediate execution mode.
5. Determining what procedures have been defined and deleting unwanted procedures.

*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

But these steps are a very small part of learning to use procedures in Logo. By using procedures a student can attack very complex problems. Problem solving is the very heart of what computers and computer programming are all about.

A student's initial introduction to writing Logo procedures must, of course, focus mainly upon the mechanics (steps 1-5 above) of the process. But very quickly you should change the focus to problem solving. Ideas such as problem segmentation and testing and debugging then become the major concerns.

The cursor control commands and other details for editing a Logo procedure vary considerably among various versions of Logo. Eventually your students will master these commands for the versions of Logo they are using. The next session is devoted to these editing features. For now, they will only be able to correct errors on the line where the cursor rests. A student's only other alternative is to delete (erase) an entire procedure. Thus, students should be encouraged to write very short procedures until they learn the editing features presented in the next session.

ACTIVITY:

Write the word "psychologist" on the board. Define the term briefly as a person who studies how people think and behave as individuals. Tell students that some psychologists observe and study how very young children learn. One psychologist thought of babies' minds as being full of blank cards in a file drawer, much like those in a library card catalog drawer. As babies learn to do things, they put the directions on one of the blank file cards. When they learn **to crawl**, they put the directions for crawling on one of the blank file cards. When babies want **to crawl**, their minds just look at the card and follow the directions on that card. (Pull out a card labeled TO CRAWL to illustrate the card concept. Call attention to the title TO CRAWL, the set of directions, and the END statement.)

Over the years, babies make many cards which tell them how to do things such as TO WALK, TO SKIP, TO JUMP, TO RIDE TRICYCLE, etc. Children make changes on the cards as they get older and learn better ways to do things. They continue to make new cards as they learn new things.

Another illustration you may want to discuss with students is that of a box of recipes. Tell students that if they wanted to bake a cake, they would look in the file card box, find the card labeled TO BAKE CAKE (pull out card), and read that set of directions.

Continue with other cards.

Tell your students that today they will be learning something new and will need a new file card. Pass out the 4x6 card for TO PALC. Read and have students follow the directions. Tell them that they now have a file card to refer to when they are asked TO PALC. Have them PALC several times in response to the word PALC (omit the word TO).

Repeat for RITONE.

Tell students they now have file cards that tell them exactly how to do two special things. They no longer have to be told *how* to do these things—they only need to read the file card and follow the listed directions. They have a *procedure* (a set of directions) to follow.

Explain that they can teach Turtle in the same way. They can make file cards so Turtle can look at the file card and follow the directions. In Logo, these file cards are called *procedures.*

Pass out the 4x6 cards for TO WELCOME and student copies of the Keyboard chart.

APPLE AND ATARI:

Tell students that one way to teach Turtle how TO WELCOME is to type TO and the name of the procedure and press <RETURN>. The computer has been programmed to give the user a "file card" whenever the word TO and the procedure name are typed and entered. Doing this puts the user into the **teaching** mode (defining a procedure), and a set of directions can be typed on one of Turtle's file cards. The last direction should be the primitive END, followed by <RETURN>.<RETURN> returns the student to the **immediate** or **command** mode.

MIT:

Tell students that one way to teach Turtle how TO WELCOME is to type TO and the name of the procedure and press <RETURN>. The computer has been programmed to give the user a "file card" whenever the word TO and the procedure name are typed and entered. Doing this puts the user into the **teaching** mode (defining a procedure), and a set of directions can be typed on one of Turtle's file cards. The last direction should be the primitive END, followed by <CTRL>C.<CTRL>C returns the student to the **immediate** mode.

TI:

Tell students that to teach Turtle how TO WELCOME, they must type TO and the name of the procedure and press <ENTER>. The computer has been programmed to give the user a "file card" whenever the word TO and the procedure name are typed and entered. Doing this puts the user into the **teaching** mode, and a set of directions can be typed on one of Turtle's file cards. END may already be on the screen. If not, it needs to be typed before returning to the **immediate** mode. <FCTN>9 (may be labeled BACK) will return the student to the **immediate** mode.

TRS:

Tell students that to teach Turtle how TO WELCOME, they must press the BREAK key, then the

E key to go into the **edit** mode. Then they must type TO and the name of the procedure and press <ENTER>. The computer has been programmed to give the user a "file card" whenever the word TO and the procedure name are typed and entered. Now a set of directions can be typed on one of Turtle's file cards. After END is entered, BREAK key, then R key will return the student to the **immediate** mode.

Demonstrate at the computer:

```
Type: TO WELCOME
      FD 70
      BK 30
      RT 90
      FD 30
      LT 90
      FD 30
      BK 70
      RT 90
      PU
      FD 30
      LT 90
      PD
      FD 70
      END
```

Now, as discussed above, show students how to return to immediate or command mode:

APPLE: <RETURN>
ATARI: <RETURN>
MIT: <CTRL> C
TI: <FCTN> <BACK>
TRS: <BREAK>, then R

Explain to students that Turtle knows how TO WELCOME because it has a file card to look at. They do not have to type in the directions each time—just as they can PALC when asked to do so, so Turtle can WELCOME. Just type the word WELCOME, not the TO. The word WELCOME is now called a command that you can give to Turtle.

First clear the screen.

Type: WELCOME (Press <RETURN> or <ENTER> .)

Allow several students to clear the screen and type WELCOME.

Add the new primitives to the Primitive Commands wall chart.

ASSIGNMENT:

Have students write a procedure on a file card giving Turtle directions for making a shape. This set of directions is called a *procedure*. They may choose any shape or design. Remind them to name their procedures and to use the word TO. They should keep the procedures short at this time. Later, they can make longer procedures.

Have students type their procedures into the computer and try them. They should keep their finished procedures in their file folders or notebooks.

Tell them that to see a list of the other procedures that Turtle knows, they should type:

APPLE: POTS (stands for Print Out TitleS)
ATARI: POTS
MIT: POTS
TI: PP (stands for Print Procedures)
TRS: <SHIFT>↑ (Edit Mode)

If other students have taught Turtle different words, they will be listed on the screen. This gives the user a list of words like a table of contents. It seems appropriate to have a class discussion about whether students should ask permission to use each other's procedures. Who "owns" programs or procedures? Is permission needed or not? Do you ask for permission to use someone else's story? Coat? What is a copyright? During the discussion, the group may reach an agreement about a classroom policy for use of personal procedures.

Tell students that the procedures will remain in the computer's memory until the computer is turned off. Later, they will learn how to save procedures when the computer is turned off. Then each procedure must have a slightly different name. You might want students to start using their initials as part of the name of the shape. For example: SRT-SQUARE or JKHSQU for a square procedure.

If students do not want to keep their procedures in the computer's memory for other students to use, they should type ERASE and the name of their procedure.

APPLE: ERASE "WELCOME
ATARI: ERASE "WELCOME
MIT: ERASE WELCOME
TI: ERASE WELCOME
TRS: Delete each character in the procedure while in the Edit mode by using <SHIFT>↑.

SUMMARY:

Summarize how to define a procedure in Logo. Have students complete the Problem Sheet.

Session 9

Procedures, Page 1

The following procedures are used in Session 9. PALC, WELCOME and RITONE are needed for each student. TO CRAWL, TO SKIP, TO BAKE CAKE and TO FIX BIKE TIRE are needed for teacher demonstration only.

```
TO PALC
STAND BY DESK
HOLD HANDS ABOUT 20 CM IN FRONT OF YOU
OPEN HANDS, PALMS FACING
QUICKLY BRING HANDS TOGETHER MAKING A
  SNAPPING NOISE
QUICKLY BRING BACK TO STARTING POSITION
REPEAT SIX TIMES
END
```

```
TO WELCOME
FD 70
BK 30
RT 90
FD 30
LT 90
FD 30
BK 70
RT 90
PU
FD 30
LT 90
PD
FD 70
END
```


© ICCE 1984. Reproduction for classroom use permitted.

```
TO CRAWL
BEGIN ON KNEES
HANDS ON FLOOR, PALMS DOWN
LEFT HAND, RIGHT KNEE FD 1
RIGHT HAND, LEFT KNEE FD 1
REPEAT PREVIOUS TWO LINES
STOP WHEN AT DESIRED PLACE
END
```

```
TO BAKE CAKE
TAKE OUT BOWL, MIXER, CAKE MIX
FOLLOW MIX DIRECTIONS
PUT IN HOT OVEN
WAIT EXACT TIME
REMOVE FROM OVEN
COOL
FROST
END
```

© ICCS 1984. Reproduction for classroom use permitted.

```
TO RITONE
PUT PENCIL ON DESK
PUT PAPER ON DESK
SIT AT DESK
PICK UP PENCIL, LEAD DOWN
PUT PENCIL DOWN ON PAPER, LEAD DOWN
SLOWLY PULL PENCIL 3 CM TOWARD LOWER
  EDGE OF PAPER
PENCIL UP
END
```

```
TO SKIP
STAND
RIGHT FOOT FD 1
RIGHT FOOT HOP
LEFT FOOT FD 1
LEFT FOOT HOP
REPEAT GAINING SPEED UNTIL SMOOTH PACE
  REACHED
END
```

© ICCE 1984. Reproduction for classroom use permitted.

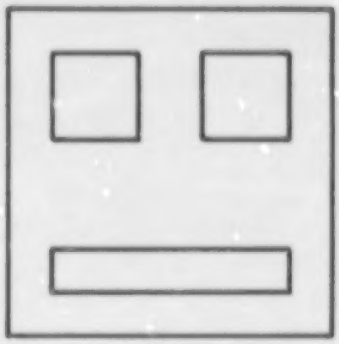
```
TO FIX BIKE TIRE
TURN BIKE UPSIDE DOWN
REMOVE WHEEL FROM FRONT FORK
REMOVE TIRE AND TUBE FROM WHEEL
FIND LEAK IN TUBE
PATCH
RESTORE TUBE AND TIRE TO WHEEL
PUMP AIR IN TUBE
PUT WHEEL BACK ON BIKE
TURN BIKE RIGHT SIDE UP
END
```

© ICCE 1984. Reproduction for classroom use permitted. 80

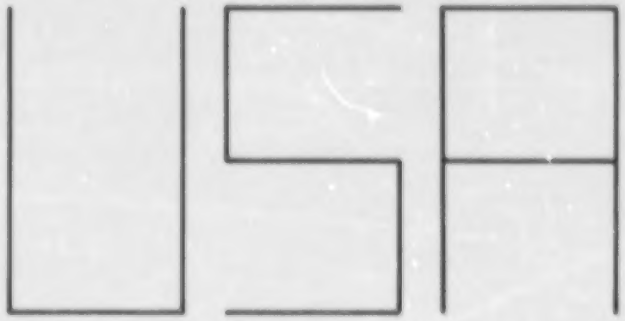
Session 9
Problem Sheet

Name _____
Date _____

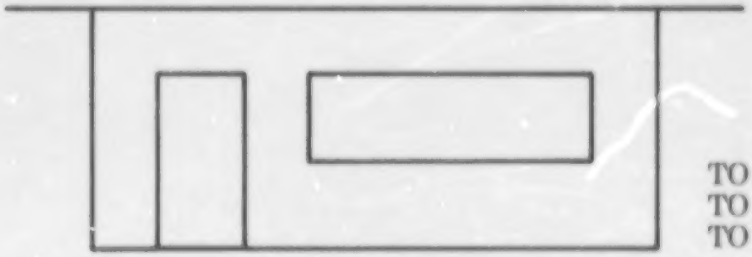
Write a procedure (a set of directions) for ONE of the designs below, or try your own initials. You may use your measuring instruments if desired. However, the designs need not be the exact same size as on this sheet. To keep the procedures from becoming too long, make separate procedures for each part of the shape. To move Turtle without a line, you will need to use PU (PENUP) and PD (PENDOWN).



TO FACE
TO EYES
TO MOUTH



TO U
TO S
TO A



TO HOUSE
TO DOOR
TO WINDOW

TO L
TO O
TO G
TO O

© ICCE 1984. Reproduction for classroom use permitted.

Session 10

*"For me, writing means making a rough draft and refining it over a considerable period of time. My image of myself as a writer includes the expectation of an 'unacceptable' first draft that will develop with successive editing into presentable form. But I would not be able to afford this image if I were a third grader. The physical act of writing would be slow and laborious. I would have no secretary. For most children rewriting a text is so laborious that the first draft is the final copy, and the skill of rereading with a critical eye is never acquired. This changes dramatically when children have access to computers capable of manipulating text."**

Seymour Papert
Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVES: To introduce the **editing mode** and give students practice in editing procedures.

CONCEPTS/SKILLS: Use of edit mode; moving cursor right, left, up and down; deleting and inserting characters and lines.

FACILITIES/MATERIALS: Chalkboard or overhead; graph paper; computer with Logo cartridge, disk or module; student copies of the Problem Sheet.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)

Defining and Editing Procedures, Managing Workspace

TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E

COMMENTS:

Editing—inserting, deleting or correcting—is a key part of writing. This is true whether you are writing an essay, a poem or a Logo procedure. The analogy between editing a pencil-written essay and a Logo procedure that has been keyed into a computer is useful. For example, suppose you want to correct the spelling of a word. One way to do this, both on paper and in Logo, is to erase the word and replace it by a properly spelled word.

But suppose one wants to make a correction that involves deleting a whole line. On paper this leaves a messy gap, but that concern is no problem in

*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

Logo. And suppose you want to insert several words or a line into the middle of some writing. In pencil-written material, this generally means recopying the entire piece. Clearly, editing on a computer is much easier than editing on paper.

Of course, students are familiar with using an eraser or recopying their writing. The initial effort to learn to edit Logo procedures may be substantial for some students. The purpose of this session is to introduce the procedure-editing facilities of Logo. These vary among different versions of Logo.

A good way to learn to edit Logo procedures is to make changes to a procedure that someone else (perhaps the teacher) has already typed in. Or, the student can be asked to type in a short procedure, execute it, modify it, execute it, modify it, etc. The advantage of the former approach is that it saves student typing time and insures the student starts with a correct procedure.

ACTIVITY:

Part 1

Use an overhead or chalkboard to display several sentences that require editing. Pick material suitable to your students' reading level.

Example:

It was a lasy day. The sun was hot, the sky was blue and a few clouds. Pat and I played ball for a hour. But then I became very very very hungary.	Replace "s" with "z" in lazy. Insert "could be seen" after clouds. Replace "a" with "about an". Delete "very very". Delete "a" in "hungary".

Illustrate how editing involves locating a spot that needs editing and then operations such as replacement, insertion and deletion.

Part 2

Give each student a piece of graph paper and put the following Logo procedure on the overhead or on the chalkboard:

```
TO FLAG
FD 60
RT 90
FD 40
RT 90
FD 20
RT 90
FD 20
END
```

Have your students carry out the procedure on their graph paper. The correct result is a flag-like shape but with a hole in one edge.

Ask your students questions such as the following:

- Suppose line 4 is changed to FD 20. What figure does the procedure draw? (Change the line on the overhead or chalkboard, and have students change their drawings.)
- Suppose next that line 8 is changed to FD 40. What figure does the procedure draw?

(Note: The idea is to have your students repeatedly follow the directions in the Logo procedure, with the changes being cumulative.)

Give them a half-dozen additional changes, including an insertion and a deletion. Be sure to change the procedure on the overhead or chalkboard and have students change their drawings.

Part 3

Discuss and illustrate the procedure-editing features of the Logo version your students are using. You might want to enter the original procedure from Part 2 onto the computer and make the same changes using the editing commands.

To Correct Errors

APPLE:

Type EDIT "_____ (procedure name). In the edit mode, the cursor is moved to the right using the right pointing arrow, to the left by using <CTRL>B; down one line by using <CTRL>N and up one line by using <CTRL>P.

- To remove an unwanted line: Move the cursor to the beginning of the line and press <CTRL>K.
- To open a space to insert a new line: Move the cursor to the beginning of the line where the new line will be inserted. Press <CTRL>O.
- To erase a character: Move the cursor so it is on the character to be erased. Press <CTRL>D.
- To add a character that was omitted: Put the cursor on the character to the right of where the new character is to be added. Type the new character.

ATARI:

Type EDIT "_____ (procedure name). In the edit mode, the cursor is moved in the direction you desire (to the right, left, up or down) by the four arrow keys.

- To remove an unwanted line: Use <CTRL>A to move the cursor to the beginning of the line. Press <SHIFT><DELETE BACK S>
- To open a space to insert a new line: Move the cursor to the beginning of the line where the new line will be inserted. Press <CTRL><INSERT>.
- To erase a character: <DELETE BACK S> erases the character to the left of the cursor. <CTRL> <DELETE BACK S> erases the character under the cursor.

- To add a character that was omitted: Put the cursor on the character to the right of where the new character is to be added. Type the new character.

MIT:

Type EDIT _____ (procedure name). In the edit mode, the cursor is moved on the screen using the arrow keys for right or left moves. Use<CTRL>N to move the cursor down one line and<CTRL>P to move it up one line.

- To remove an unwanted line: Move the cursor to the beginning of the line and press<CTRL> K.
- To open a space to insert a new line: Move the cursor to the beginning of the line just below where the new line will be inserted. Press <CTRL>O.
- To erase a character: Move the cursor so it is on the character to be erased. Press<CTRL>D.
- To add a character that was omitted: Put the cursor on the character to the right of where the new character is to be added. Type the new character.

TI:

Type TO _____ (procedure name). In the teaching mode, the cursor is moved on the screen by holding the<FCTN>key down and using the arrow keys.

- To remove an unwanted line: Move the cursor to the beginning of the line and press< FCTN> <CLEAR>.
- To open a space to insert a new line: Move the cursor to the end of the line and press <ENTER>.
- To erase a character: Use the arrow keys to put the cursor on the character to be erased. Then, press<FCTN>< DEL>.
- To add a character that was omitted: Put the cursor on the character to the right of where the new character is to be added. Type the new character.

TRS:

Press E to enter the edit mode from the BREAK mode. The cursor is moved in the direction you desire using the arrow keys. (↑ moves the *text* up one line. ↓ moves the *text* down one line.)

- To open a space to insert a new line: Move the cursor to the beginning of the line where the new line will be inserted. Press<SHIFT>↓.
- To erase a character: Move the cursor over the character to be erased and press<SHIFT>←. (An unwanted line may be erased one character at a time using the<SHIFT>←keys.)
- To add a character that was omitted: Put the cursor on the character to the right of where the new character is to be added. Press<SHIFT>→. Now type the new character.

ASSIGNMENT:

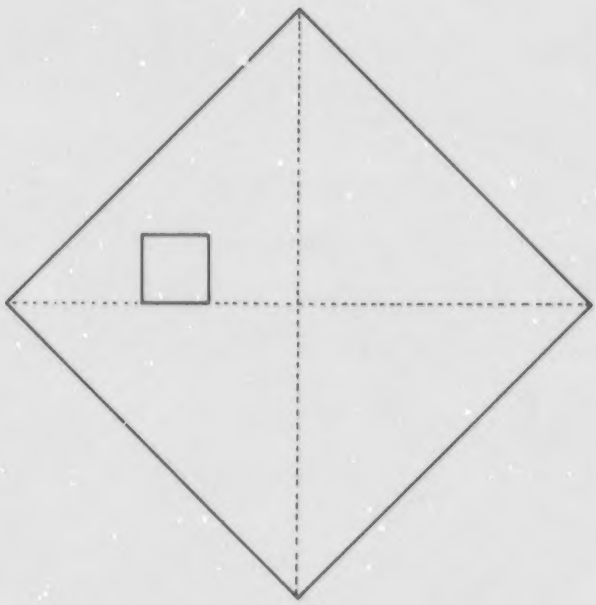
Have students sign up for time on the computer to practice editing Logo procedures. You may wish to have students make up their own practice materials or to edit a procedure that has been typed in for them. Or you may have students complete the Problem Sheet which contains a procedure for you to enter and have students edit. (Note: You will need to enter the procedure, save it on a disk or cassette, and load it for students at the beginning of computer time.)

Session 10 **Name** _____

Problem Sheet, Page 1 **Date** _____

1. Type the command PLAY. Your teacher has entered the procedures into the computer. It should draw a figure with a small square inside.

```
TO PLAY
RT 45
FD 60
RT 90
FD 60
RT 90
FD 60
RT 90
FD 60
RT 135
PENUP
FD 20
PENDOWN
FD 10
LT 90
FD 10
LT 90
FD 10
LT 90
FD 10
END
```



2. Make the following changes to the procedure. After each change clear the screen and execute the procedure. Sketch the resulting figure on this problem sheet.

a. Change line 1 from TO PLAY
to TO (your name). Sketch:

© ICCE 1984. Reproduction for classroom use permitted.

Session 10
Problem Sheet, Page 2

Name _____
Date _____

b. Change line 2 to LT 45. Sketch:

c. Change line 12 to BK 20. Sketch:

d. Change line 12 to FD 20. Sketch:

e. Delete line 13. Sketch:

f. Insert the line you deleted in (e.) into the place it was originally. Sketch:

g. Experiment with additional changes of your choice. Sketch:

© ICCE 1984. Reproduction for classroom use permitted.

Session 11

*"Errors benefit us because they lead us to study what happened, to understand what went wrong, and, through understanding, to fix it. Experience with computer programming leads children more effectively than any other activity to 'believe in' debugging. . . . you can figure out 'why the Turtle did that dumb thing' by following through on [an] already well-known piece of heuristic advice: Play Turtle. Do it yourself but pretend to be as dumb as the Turtle. Finding out why the Turtle did it almost immediately suggests a way to fix it."**

Seymour Papert
Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVES: To introduce the word "debugging" to refer to finding and correcting errors in procedures, and to give students practice in using the editing features in Logo.

CONCEPTS/SKILLS: Locating and correcting errors in procedures and understanding error messages in Logo.

FACILITIES/MATERIALS: Overhead, chalkboard or computer; dictionary; student copies of the two mazes and two Problem Sheets.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)

Defining and Editing Procedures, Managing Workspace

TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E

COMMENTS:

Soon after the first computers were built, a moth got into one and shorted out a connection. This brought the room-size machine to a halt. From this story came the idea of a bug (in computer hardware) and debugging. Eventually, the term "bug" came to mean an error in a program. Debugging refers to detecting and correcting bugs in a program.

Everyone who writes computer programs is faced by the tasks of program testing, detecting bugs and correcting bugs. This is a natural and everyday part of developing and writing procedures to solve problems. While debugging sometimes carries a slightly negative connotation, for the most part, it is viewed

*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers.

as a positive and important part of the overall programming process.

Contrast this with the attitude that most students develop toward making a mistake in doing arithmetic, artwork or writing. Most students are taught (or somehow come to accept) the idea that to err is bad; an error or mistake carries a serious negative connotation.

Seymour Papert talks about "powerful ideas" and the use of Logo to help students internalize these powerful ideas. Bugs and debugging, viewed positively and as an important aspect of problem solving, constitute a powerful idea. As you help your students to learn about bugs and debugging, it is very important that you stress the positive aspects of the overall process.

A Logo procedure is designed to solve a problem or a piece of a problem. Typically a student's first effort in attacking a problem will result in procedures that contain bugs. That is, the student will produce procedures that are related to the problem and are a step toward solving the problem, but are flawed in some manner. The bugs may be divided into two major categories:

1. Syntax errors (violations in the basic rules of Logo). Examples:

   FD35 (no blank space after FD)
   PANUP (should be PENUP)
   RT (RT needs to be followed by a value)

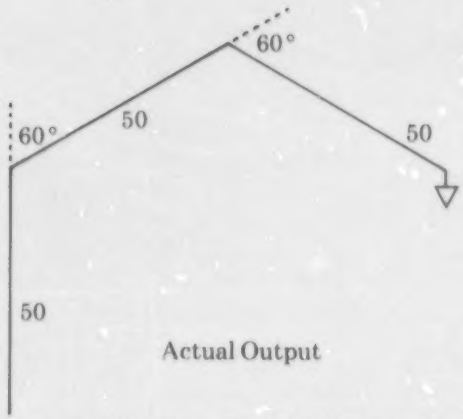
   The Logo system detects such errors and prints out error messages to help the programmer understand what is wrong. Generally speaking, detecting and correcting syntax errors is quite easy.

2. Logic errors (the program runs all right, but fails to produce the desired results). Example:

   ```
   TO TRIANGLE
   FD 50
   RT 60
   FD 50
   RT 60
   FD 50
   RT 60
   END
   ```

   The writer of this procedure had in mind drawing an equilateral triangle with sides of length 50. Each angle in such a triangle is 60°.

Desired Output



Actual Output

The power of Logo as an aid to learning is most vividly illustrated as you work to detect and correct bugs in procedures that direct Turtle to draw a picture. The student can detect that there is a bug—Turtle does not follow the path the student had in mind. The student can search for the precise reasons why Turtle deviates from the desired path. Playing Turtle—mentally, on a piece of paper or walking out the procedure steps—is the standard way to search for a bug. Once a bug is detected, the student can edit the procedure and execute it again. The student learns to become self-reliant in detecting and correcting bugs.

Mentally or physically walking the commands is often called tracing through the program's execution. It is important to encourage students to do a trace when they have a bug in their procedure. Some students may try a trial-and-error approach—randomly changing something and then trying the procedure to see if the change helps. If students always take this approach to debugging, they will never understand how and why their procedures work even though the procedure may accomplish the desired result.

ACTIVITY:

Use the two mazes included with this session to introduce the concept of a "bug" as a block that interferes with a chosen path. Let students draw their route on the maze with the bugs. Then tell them to draw a path through the maze with the option of removing the bugs or drawing through any bugs in their pathway.

Next, give students the second maze. This maze is the exact copy of the first but does not have the bugs. This maze gives the option of a straight route through the maze—amazingly easy with the bugs removed.

Ask older children to write a procedure for the path on each maze using protractors and metric rulers for measuring instruments. (The procedure for the bug-free maze will be a one-line procedure with a FORWARD command.)

Explain to students that sometimes bugs are hidden in procedures, and the computer is blocked from completing the set of directions until the bugs are removed. When we write procedures, sometimes we make the wrong turn, forget a space, or spell a word incorrectly. Problems in a procedure that cause trouble for Turtle are called *bugs*. Or, if the result of your procedure is not what you want, then the procedure contains a bug.

Show students the five words below, using the overhead or chalkboard. Discuss the meanings. Some students might also locate the words in the dictionary.

de-ice	take ice away or out of
deflate	let air out of
defrost	to melt
degrease	take the grease from; ungrease
deodorize	to take the smell out of or away from, as in deodorizing a skunk for a pet

Ask students what we could call taking the bugs out of procedures. Lead them to suggest "debug" or "debugging." Then explain that this is the computer term for correcting errors.

Illustrate on the overhead or computer some possible errors. If you use the computer, you will be able to see and read the error messages printed on the screen by the computer.

1) ERROR: F20
   - What is the mistake?
   - How can it be corrected?

   (D in FD omitted. Also, a space needed between command and input.
   (The computer will tell the user that F20 is not one of the words or primitives that it knows. Of course, if someone would teach it to F20, it could then carry out that command.)

2) ERROR: RT
   - What is the mistake?
   - How can it be corrected?

   (Number left out)

3) ERROR: TO BACK
   - What is the mistake?
   - How can it be corrected?

   (Will not allow the user to write a procedure with this name since BACK is one of the primitives. Note: TRS Logo gives no error message in this case.)

4) ERROR: FD
   - What is the mistake?
   - How can it be corrected?

   (Turtle is asking for more input—a number with the FD command.)

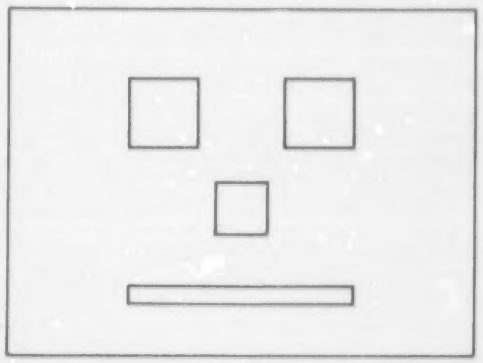
To Correct Errors:

First, you must find the bug in the procedure. Sometimes this can be done by just reading through the procedure. If the bug is not easily found, tell students to try playing Turtle. One student can read the set of directions as another student pretends to be Turtle and walks out the commands. Sometimes having more than one classmate join in the fun of playing Turtle can help find the bugs easily.

Once the bug has been located, return to the editing mode for correcting the procedure.

ASSIGNMENT:

1. Have students complete the Problem Sheet.
2. Suggest that students create and debug a procedure of their own choice. Alternatively, drawing a robot face is a good challenge. A simple one might look as follows:

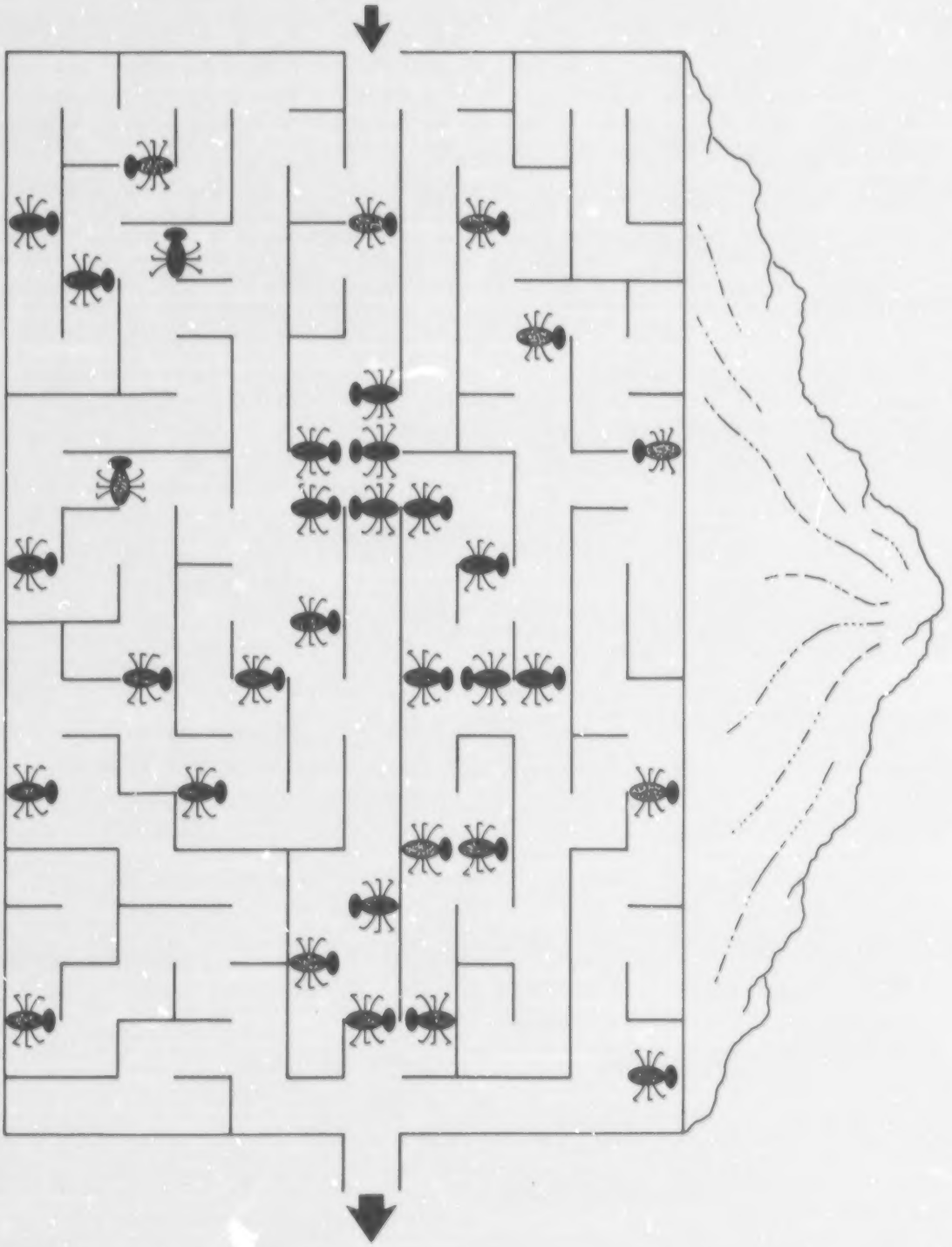


More advanced students can work on much more complex faces.

Session 11
Maze

Name _____
Date _____

"Don't Bug Me"

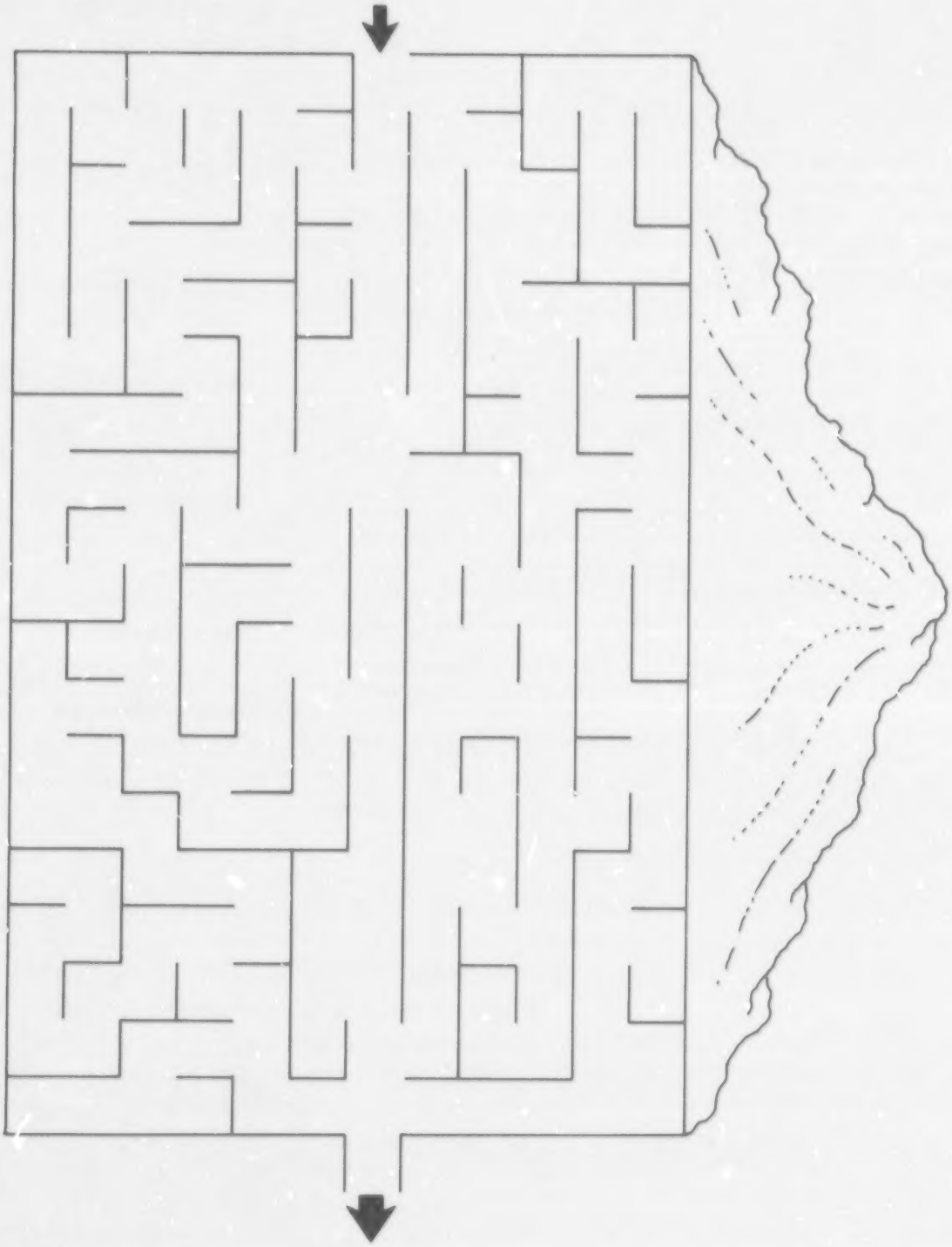


© ICCE 1984. Reproduction for classroom use permitted.

Session 11
Maze

Name _____
Date _____

"Bug Free"



© ICCE 1984. Reproduction for classroom use permitted.

Session 11 **Name** _____

Problem Sheet I **Date** _____

You may work with a partner on this assignment. Debug the following procedures. Make the corrections and try on the computer. If you have difficulty, PLAY TURTLE!

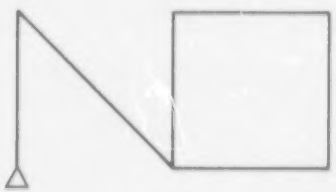
```
TO FLAG
FD 30
RT 90
FD 30
LT 90
FD 30
RT 90
FD 30
LT 90
FD 30
END
```

I wanted:

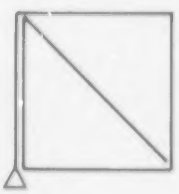
I got:

```
TO NO
FD 30
LT 45
BK 40
RT 45
FD 30
LT 90
FD 30
LT 90
FD 30
LT 90
FD 30
LT 90
END
```

I wanted:



I got:



(Turtle is at starting position.)

How can you debug the following?

1. Enter this in the computer: TO HOME
   What does the computer print? _____
   How can you solve this problem? _____
   (Note: TRS Logo gives no error message.)

2. Enter this in the computer: FD #
   What does the computer print? _____
   How can you solve this problem? _____

3. Enter this in the computer: AHEAD 20
   What does the computer print? _____
   How can you solve this problem? _____

© ICCE 1984. Reproduction for classroom use permitted.

Session 11 **Name** _____

Problem Sheet II **Date** _____

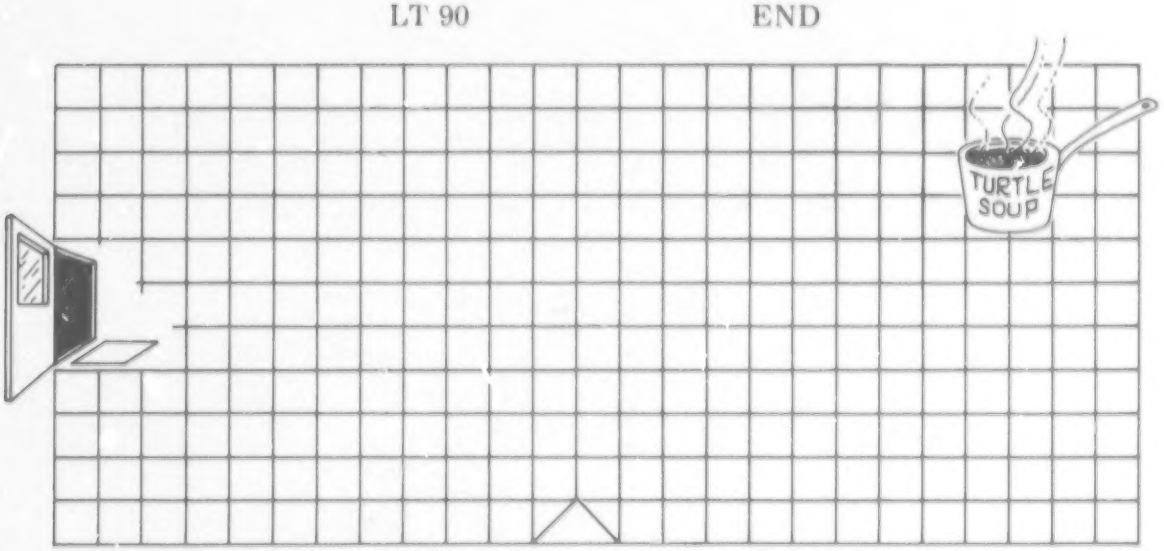
"Help Turtle Have Dinner with Hungry Cat"

Hungry Cat decided to cook a fancy dinner and invite a friend. HC sent the following directions telling Turtle how to get to his house. While HC was writing, a little bug bumped his arm making HC write one wrong turn. Draw the path and find how Turtle arrived at the wrong place. Hungry Cat must have sent the wrong directions to Turtle! This looks like Turtle Soup!

Can you find the place in the procedure where the bug gave *one wrong turn* to Turtle? *Change that line.* Then draw Turtle's new path so HC can enjoy dinner *with* Turtle instead of enjoying Turtle. If Turtle's path leaves the grid it cannot return. Turtle is lost in outer space! The path must go *through the door.* You may work with a friend or in a small group.

TO VISIT HC'S HOUSE

FD 4	FD 7
LT 90	LT 90
FD 7	FD 4
RT 90	LT 90
FD 2	FD 8
RT 90	LT 90
FD 6	FD 4
LT 90	RT 90
FD 2	FD 10
LT 90	END



© ICCE 1984. Reproduction for classroom use permitted.

Session 12

"Total turning is the central concept here. . . . One can imagine any turtle program keeping a running count of its turning, adding in RIGHTs and subtracting LEFTs. Because only RIGHTs and LEFTs change heading, this total turning is always exactly the total change in heading. In particular, if the path is a **closed path** *(one which restores the turtle's initial position and heading), we can be confident that the net turning (= changing of heading) is a multiple of 360°. This gives us our first turtle-geometric theorem:*

"CLOSED-PATH THEOREM: The total turning along any closed path is an integer multiple of 360°. . . . That integer is called the rotation number of the path.

"SIMPLE-CLOSED-PATH THEOREM: The total turning in a simple closed path is 360° (to the right or to the left). This is to say, the rotation number of any simple closed path is ± 1*."**

Harold Abelson and Andrea di Sessa
Turtle Geometry, The Computer as a Medium for Exploring Mathematics

OBJECTIVES: To introduce the concept of one rotation as an amount of turning which is equal to 360°, and to reinforce the concept of procedures as a means of teaching Turtle to follow a set of instructions.

CONCEPTS/SKILLS: Degrees, rotation and procedure.

FACILITIES/MATERIALS: (For Part 1) For each student: calculator, large protractor and preprinted 4x6 cards with the procedures TO GET READY and TO ROTATE. (For Part 2) Gym, hallway, playground or sidewalk. For each pair of students or small group: appropriate medium for drawing shapes (water-base markers, chalk, sticks to scratch lines in sand, tempera paint to sprinkle on snow or yarn/crepe markers for grass), large protractors, and meter sticks. For Assignment: Student copies of the Problem Sheet and the Challenger Sheet. (See the ASSIGNMENT section to determine which sheets are appropriate.)

PREPARATION: A master for the two procedures TO GET READY and TO ROTATE is included at the end of this session. The master can be duplicated on tagboard/construction paper or photocopied and mounted on tagboard and then cut into 4x6 cards.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC) COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)
HIDETURTLE (HT)	HT	HIDETURTLE (HT)	HIDETURTLE (HT)	HIDETURTLE (HT)
SHOWTURTLE (ST)	ST	SHOWTURTLE (ST)	SHOWTURTLE (ST)	SHOWTURTLE (ST)

*Cambridge, Massachusetts: The MIT Press, 1980. © 1980 by The Massachusetts Institute of Technology. All rights reserved. No part of this book may be reproduced or by any means, electronic or mechanical, including photocopying, recording, or by an information storage and retrieval system, without permission in writing from the publisher.

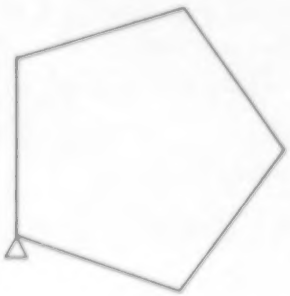
TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E

COMMENTS:

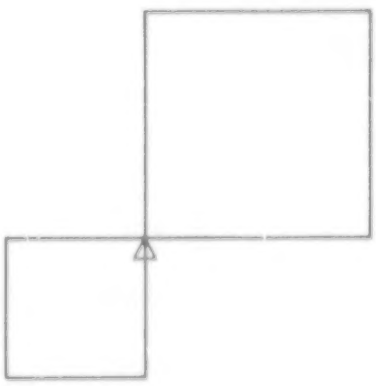
Study the procedures and figures given below. Each procedure directs Turtle along a path and back to its original starting point and heading.

```
TO SQUARE
LT 90
FD 50
LT 90
FD 50
LT 90
FD 50
LT 90
FD 50
END
```

```
TO PENTAGON
FD 50
RT 72
FD 50
RT 72
FD 50
RT 72
FD 50
RT 72
FD 50
RT 72
END
```



```
TO TWOBOX
FD 50
RT 90
FD 50
RT 90
FD 50
RT 90
FD 50
LT 90
FD 30
RT 90
FD 30
RT 90
FD 30
RT 90
FD 30
LT 90
END
```



In each procedure suppose that RT is considered as a turn in a positive direction and LT as a turn in a negative direction. Then you can add up all of the turns in any one of the procedures. For example, in SQUARE the result is $(-90°) + (-90°) + (-90°) + (-90°) = -360°$. In PENTAGON it is $(+72°) + (+72°) + (+72°) + (+72°) + (+72°) = +360°$. Try adding the turns for TWOBOX.

Notice that in all three examples, the sum of all the turns is an integer multiple of 360°. This is not a coincidence. Rather, it can be mathematically proven that this result holds for every Turtle path that returns to its original starting point and starting heading. This is a beautiful mathematical result which you can help your students to discover and explore.

It is also a very useful result. In a previous session we looked at the following procedure.

```
TO TRIANGLE
FD 50
RT 60
FD 50
RT 60
FD 50
RT 60
END
```

The procedure was designed to draw a triangle with Turtle returning to its initial starting point and heading. But notice that the total turning is $(+60°) + (+60°) + (+60°) = +180°$. Since 180° is not an integer multiple of 360°, we know (due to the mathematics of Turtle geometry) that the procedure cannot return Turtle to its original starting point and heading.

This session is a blend of mathematics and Logo. It involves some long division. For example, suppose a student wants to make an 8-sided regular polygon. Then turns of $360°/8 = 45°$ are required at the end of each side. If your students can't handle such division problems, you may want to supply them with calculators and/or teach them to use the Logo system as a calculator.

ACTIVITY:

Part 1

Pass out the large protractors. Have students stand and face the north wall of the classroom. Using the large protractors, have students make a 90°

right turn and observe that they are now facing east. Have them make a 90° left turn and observe that they are now back to their original heading (that is, facing north).

- Make a RT 100—return to starting heading.
- Make a LT 60—return to starting heading.

Questions to ask, all starting with a north heading:

- When you turn RT 90, which direction are you facing? (east)
- When you turn LT 90, which direction are you facing? (west)
- When you turn RT 180, which direction are you facing? (south)
- When you turn LT 180, which direction are you facing? (south)
- After a LT 75, what turn is needed to return to a north heading? (RT 75; note, of course, that answers such as LT 285 and RT 435 are also correct.)

Next, you want to work on the idea that there are 360° in a complete rotation. Pass out the cards TO GET READY and TO ROTATE. Allow time for students to read the cards, then guide them through the directions. Practice first using GET READY and then ROTATE.

Have students ROTATE again, but this time count the number of times they move their right foot. Ask if they can rotate making the rotation in exactly four equal turns. Help them to discover that each of these turns is 90° and that four turns of 90° (a total of 360°) makes one rotation.

You might want to put the following calculations on the chalkboard or overhead projector.

$360°/4=90°$ $4x90°=360°$
$360°/8=45°$ $8x45°=360°$

Then ask questions such as:

- How many RT 90 turns are required for one complete turn? (4)
- How many LT 90 turns are required for one complete rotation? (4)
- How many degrees in one complete rotation? (360°)
- How many RT 45 turns are required for one complete rotation? (8)
- How many LT 45 turns are required for one complete rotation? (8)
- What will Turtle do if you type RT 360? LT 360?
- What will Turtle do if you type the sequence:

  RT 90 RT 45 RT 45 RT 90 RT 90
- Suppose one complete rotation is made in five equal turns. How many degrees in each turn? (360°/5=72°).
- Suppose one complete rotation is made in six equal turns. How many degrees in each turn? (360°/6=60°)

Part 2

In Part 1, students discover that making one complete rotation without any forward movement requires turning 360°. In Part 2, students explore making a sequence of equal movements and turns. For example, four repetitions of FD 50 RT 90 draws a square and returns Turtle to its original position and heading. Five repetitions of FD 50 RT 72 draws a pentagon. Five repetitions of RT 72 FD 50 also draws a pentagon. This general idea allows one to direct Turtle to draw any regular polygon.

This activity requires a spacious area (gym, hallway, playground, sidewalk). Provide an appropriate drawing medium for the surface, large protractors and meter sticks (see Facilities/Materials).

Divide students into pairs or small groups and assign one number to each group (45, 60, 72, 90 or 120). This assigned number is the number of degrees in each turn they will make. Have each group locate a starting point and direction and choose a fixed distance. The procedure is to use the large protractor to turn right or left the assigned number of degrees and then use the meter stick to measure and draw a path the chosen fixed distance. The path will be drawn using the appropriate medium for the surface. Students should keep track of the total number of degrees turned. This turning and measuring process continues until students end up at their starting point facing their starting direction. The figure created should be one of the regular polygons and the total number of degrees turned should be 360°.

COMPUTER ASSIGNMENT:

Have students write a set of directions—a procedure—to teach Turtle to make a complete rotation. In between turns, tell Turtle to go FD 15.

```
EXAMPLE: TO POLY
         FD 15
         RT 60
         FD 15
         RT 60      (In long repetitious procedures, leaving every
                    5th line blank aids
         FD 15      readability on paper.)
         RT 60
         FD 15
         RT 60

         FD 15
         RT 60
         FD 15
         RT 60
         END
```

Have students type the procedure into the computer to see if Turtle will make a complete rotation and end in its starting position. Students should complete the Problem Sheet and the Challenger Sheet, if appropriate.

The Challenger Sheet is meant *only* for those students who have a good understanding of the concept of 360° in one rotation. The Challenger Sheet begins with the concept of 360° in one rotation; each additional rotation adds in units of 360. It then takes the more advanced student into the concept of THE CLOSED PATH THEOREM: **The total turning along any closed path is an integer multiple of 360°**. The Turtle or student must end facing the same direction as it was when it began making its path.

EXTENSION:

As an art activity, you may want to suggest that students try using the polygon procedures with a variety of colors to make colorful designs. This can be done on the computer or on art paper for display.

As an extra activity, have students draw paths in the gym or make snow paths. Make several that close. (They need not be regular polygons.) Measure the number of degrees at each turn. The total number of degrees should equal 360°.

Use a city map. Draw a tour of the town. Measure the total turning required if returning to the original point of departure and original heading. Add right turns, subtract left turns.

```
TO GET READY
STAND BY YOUR DESK
PUT HANDS BY YOUR SIDES
FACE NORTH
END
```

```
TO ROTATE
PICK UP RIGHT FOOT
TURN TOE TO YOUR RIGHT
PUT RIGHT FOOT DOWN
PICK UP YOUR LEFT FOOT
PUT LEFT FOOT DOWN BY RIGHT FOOT
REPEAT UNTIL BACK IN STARTING POSITION
END
```

© ICCE 1984. Reproduction for classroom use permitted.

Session 12
Problem Sheet

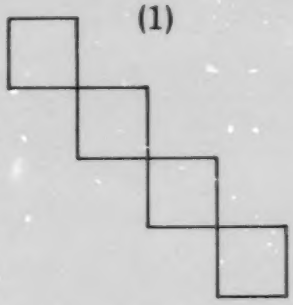
Name _____

Date _____

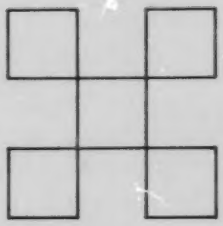
Write a procedure (a set of directions) to make any ONE of the following designs. You may use any of the following procedures *in your procedure* or you may make your own. You may want to use PU and PD.

```
TO SQUARE
FD 15
RT 90
FD 15
RT 90
FD 15
RT 90
FD 15
RT 90
END
```

(1)

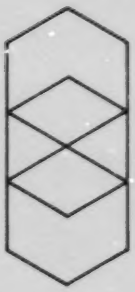


(2)

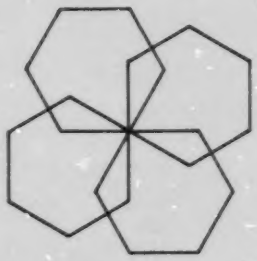


```
TO HEX
FD 15
RT 60
FD 15
RT 60

FD 15
RT 60
FD 15
RT 60

FD 15
RT 60
FD 15
RT 60
END
```

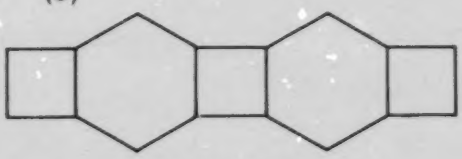
(3)



(4)

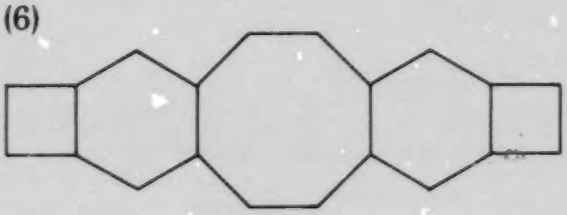


(5)



```
TO OCT
FD 15
RT 45
FD 15
RT 45
FD 15
RT 45

FD 15
RT 45
FD 15
RT 45
FD 15
RT 45
END
```

(6)



TRY TELLING TURTLE HT AND ST !!??!!

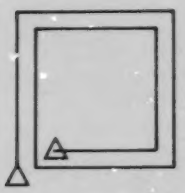
© ICCE 1984. Reproduction for classroom use permitted.

Session 12
Challenger Sheet

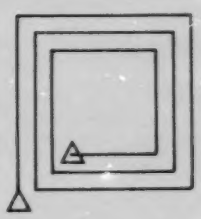
Name _____
Date _____

Answer the following questions using the figures at the left of each set of questions.

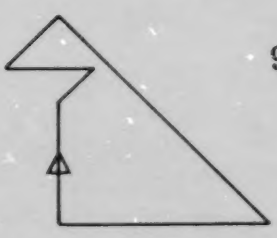
1. Each turn required how many degrees? _____
2. How much turning did Turtle do to make this shape and again be headed north? _____



3. Each turn required how many degrees? _____
4. How much turning did Turtle do to make the shape and again be headed north? _____
5. How many complete rotations were required? _____



6. Each turn required how many degrees? _____
7. How much turning did Turtle do to make the shape and again be headed north? _____
8. How many complete rotations were required? _____



9. Measure each angle. Add all the **right** turns, and subtract all the **left** turns. Total = _____.

10. A regular six-sided figure (hexagon) requires 360° turning (6 sides x 60°). A regular five-sided figure (pentagon) requires 360° total turning (5 sides x 72°). At the computer, have Turtle draw a figure in which it goes FD 30 and RT 144. How many times do you have to repeat this to have the figure close (so Turtle is back in its starting position)? _____

    What is the total turning? _____

11. Draw a new figure using FD 30 and RT with another angle. Will it close? What is the total turning? _____

© ICCE 1984. Reproduction for classroom use permitted.

Session 13

"Only humans and computers store information outside of themselves."
Dick Ricketts

OBJECTIVES: To introduce saving and recalling of Logo procedures; to help students develop an understanding of the computer's process of recording information on cassettes or disks.

CONCEPTS/SKILLS: Secondary storage.

FACILITIES/MATERIALS: Overhead or chalkboard; transparency with a 10x28 grid or a 10x28 block grid on chalkboard; student copies of 10x28 grid; computer and tape recorder with cassette or disk drive with disk; student copies of Problem Sheet; and TI users: PUTTILE wall chart.

PREPARATION: A master for preparing an overhead transparency and student copies of the 10x28 grid is included before the Problem Sheet. See your computer manual for complete details for secondary storage on your particular computer. TI users: A master for the PUTTILE wall chart is in the Charts section. Display after students have completed the Problem Sheet.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)
HIDETURTLE (HT)	HT	HIDETURTLE (HT)	HIDETURTLE (HT)	HIDETURTLE (HT)
SHOWTURTLE (ST)	ST	SHOWTURTLE (ST)	SHOWTURTLE (ST)	SHOWTURTLE (ST)
FULLSCREEN	FS	FULLSCREEN		
SPLITSCREEN	SS	SPLITSCREEN		
			NOTURTLE	
			PUTTILE (PT)	
				SLOW

Defining and Editing Procedures, Managing Workspace and Files

TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E
SAVE	SAVE	SAVE	SAVE	S
LOAD	LOAD	READ	RECALL	L
ERALL	ERALL	ERASE ALL		

COMMENTS:

Most microcomputer systems have two types of memory storage. Primary memory is limited in size and is for the temporary storage of programs and data. Primary memory is erased when the computer system is turned off.

Secondary storage involves use of magnetic disks or magnetic tape. It is usually called permanent storage, although a disk or tape will wear out eventually.

Students can become confused by the difference between primary (temporary) storage and secondary (permanent) storage. For primary (temporary) storage, the main idea to cover is that there is some temporary storage space, called the computer workspace. Both the programmer and the computer make use of this workspace. When a student writes a procedure, this procedure is (automatically) stored in the workspace.

In Session 9 we covered the POTS command for printing out the titles of procedures stored in the workspace. We also covered how to erase a procedure from the workspace. You may need to review these ideas with your students. This is a good place to discuss and/or review ethical and responsible behavior. Students should learn that they do not erase other students' procedures without appropriate permission.

Procedures and data are stored in primary or secondary storage using a binary code. Outside the machine we think of this as sets of 0s and 1s. Inside the machine these may be open/closed circuits; positive or negative voltages; magnetized or non-magnetized spots on tape or disk. The computer's circuitry automatically takes care of the process of translation between the character codes humans are used to using (A, B, C, D, etc. and 0, 1, 2, 3, etc.) and the machine representations.

This session introduces students to a binary code for characters. You might want to see if some of your students have worked with Morse code. In Morse code sets of dots and dashes (short and long beeps) are used to represent letters and digits.

This session also covers storing procedures on disk or tape and reading procedures from tape or disk into the computer workspace. This opens up a new world to students—use of a permanent library of Logo procedures. If resources permit, you will want each student to have a disk or tape for the permanent storage of procedures.

PRELIMINARY ACTIVITY:

Use a few guess-my-pattern tasks to orient the students to building number charts.

Write on board or overhead:

Ask students to provide the numbers:

```
                1
              1 2
            1 2 3
          1 2 3 4
      1 2 3 - -        (12345)
    1 2 - - - -        (123456)
  1 2 3 - - - -        (1234567)
```

and:

```
          1 2 3
        1 2 - 4        (3)
      1 - 3 4 1        (2)
    1 2 3 - 1 2        (4)
  1 - 3 4 1 2 -        (23)
1 2 3 - - - - -        (41234)
```

and:

```
          9 8 7 6
        9 8 7 6 5
      9 8 7 6 5 4
    9 8 7 6 - - -      (543)
  9 8 7 - - - - 2      (6543)
9 8 - - - - - - 1      (765432)
9 8 7 6 5 4 - - - 9    (321)
9 - - - - - - - - - -  (8765432198)
```

Discuss vertical, horizontal and diagonal patterns.

ACTIVITY:

Part 1

Tell students that they are going to make a secret code using a number chart (on a 10x28 cm grid). They will then use the code to find some new primitives.

Pass out student copies of the 10x28 cm grid.

Use the overhead grid or chalkboard grid. Have students work along with you, using their sheets of grid paper, while you use the overhead or chalkboard.

1							1	A
2						1	0	B
3						1	1	C
4					1	0	0	D
5					1	0	1	E
6					1	1	0	F
7					1	1	1	G
8				1	0	0	0	H
9				1	0	0	1	I
10				1	0	1	0	J
11				1	0	1	1	K
12				1	1	0	0	L
13				1	1	0	1	M
14				1	1	1	0	N
15				1	1	1	1	O
16			1	0	0	0	0	P
17			1	0	0	0	1	Q
18			1	0	0	1	0	R
19			1	0	0	1	1	S
20			1	0	1	0	0	T
21			1	0	1	0	1	U
22			1	0	1	1	0	V
23			1	0	1	1	1	W
24			1	1	0	0	0	X
25			1	1	0	0	1	Y
26			1	1	0	1	0	Z
27			1	1	0	1	1	

1. Number the rows from 1 to 27, using the first column on the left on the blank grid.
2. List A through Z down the last column. (The blank square at the end of the column represents a space.)
3. Tell students that we want to make a number code but all we can use are zeros and ones.

   A is 1 (add to chart)
   B is 10 "
   C is 11 "
   D is 100 "
   E is 101 "
   F is ??? "
   G is ??? "

   Continue the chart, having students give the numbers.
4. After the chart is complete, discuss the resulting patterns.
5. Write the following on the chalkboard or overhead and let students fill in the letters and blanks.

There are two messages for all Logo versions and one message for each particular version.

ALL VERSIONS OF LOGO:

10100 10101 10010 10100 1100 101

(TURTLE)

1100 1111 111 1111 11011 1001 10011 11011 110 10101 1110

(LOGO IS FUN)

FOR SPECIFIC VERSIONS:

TI:

10011 1 10110 101 11011 1 1110 100 11011 10010 101 11 1 1100 1100

(SAVE AND RECALL)

APPLE:

1011 1 10110 101 11011 1 1110 100 11011 1100 1111 1 100

(SAVE AND LOAD)

ATARI:

1011 1 10110 101 11011 1 1110 100 11011 1100 1111 1 100

(SAVE AND LOAD)

MIT:

1011 1 10110 101 11011 1 1110 100 11011 10010 101 1 100

(SAVE AND READ)

TRS:

10011 11011 1011 101 11001 11011 1 1110 100 11011 1100 11011 1011 101 11001

(S KEY AND L KEY)

The last secret message will give some new primitives!

Summarize by asking students if they could write a secret message to someone using only zeros and ones. Explain that this is the way a computer saves procedures. The letters, numbers and other characters from the keyboard are coded into ones and zeros and saved magnetically on a cassette tape or floppy disk.

Show and name the peripherals: cassette recorder, tape and cable connection; or the disk drive, controller (if one is encased as a separate unit) and a floppy disk.

Part 2

Demonstrate the saving and loading process at the computer.

1. Clear the computer workspace so there are no procedures in memory.

   APPLE: ERALL
   ATARI: ERALL
   MIT: ERASE ALL (ER AL)

   TI: ERASE _____ (procedure name) Each procedure must be erased separately. If there are many procedures in the workspace, it may be faster to type BYE and again enter Logo.

   TRS: Erasures must be done in BREAK mode. Press the two keys: <SHIFT> <CLEAR>. All the procedures in the workspace are erased.

2. Type in two or three short procedures.

   Example:
   ```
   TO TRIANGLE
   FD 40
   RT 120
   FD 40
   RT 120
   FD 40
   RT 120
   END
   ```

3. Guide the class through the complete process of saving the contents of the workspace. Choose a file name. All the procedures in the workspace will be saved under one file name such as DEMO.

   APPLE: SAVE "_____ (file name)
   Press <RETURN>.)
   Example: SAVE "DEMO

   ATARI: For Cassette:
   SAVE "C:
   (First set the tape recorder counter to 000. After typing SAVE "C: and <RETURN> the computer will beep twice. Simultaneously, press the PLAY and RECORD buttons on the tape recorder. Press <RETURN> on the computer. Advance the counter about 10 counts before saving the contents of another workspace.)

   ATARI: For Disk:
   SAVE "D:_____ (file name)
   (Press <RETURN>)
   Example: SAVE "D:DEMO

   MIT: SAVE "_____ (file name)
   (Press <RETURN>.)
   Example: SAVE "DEMO

   TI: SAVE (Press <ENTER> and follow computer prompts.)

TRS: For Cassette:
S (in BREAK mode):
(The word SAVE and the cursor appear on the screen LOGO: SAVE:__
Check the volume control setting on the tape recorder. Rewind the tape to a blank area. Note the counter. Press down the RECORD and PLAY buttons on the tape recorder at the same time. Press T (for tape), then press <ENTER> to begin recording. Allow about 10 counts between locations on the tape whenever you save the contents of another workspace.)

For Disk:
S (in BREAK mode):
(The word SAVE and the cursor appear on the screen LOGO: SAVE:__
Enter a letter A-P.)

4. Again, clear the workspace and attempt to run the procedures to demonstrate that the computer has no procedures in memory at this time.

5. Guide the class through the process of recalling the procedures from the file on cassette or disk into the computer's workspace.

APPLE: LOAD "_____ (file name)
(Press <RETURN>.)
Example: LOAD "DEMO

ATARI: For Cassette:
LOAD "C:
(First set counter to beginning number of the file you want to retrieve. Type LOAD "C: and press <RETURN>. After the computer beeps, press the PLAY button on the tape recorder and press <RETURN> on the computer.)

ATARI: For Disk:
LOAD "D:_____ (file name)
(Press <RETURN>.)
Example: LOAD "D:DEMO

MIT: READ "_____ (file name)
(Press <RETURN>.)
Example: READ "DEMO

TI: RECALL
(Press <ENTER> and follow computer prompts.)

TRS: For Cassette:
L (in BREAK mode):
(The word LOAD and the cursor appear on the screen LOGO: LOAD:__
Locate the point on the tape, using the tape recorder counter, where the file was recorded. Check the volume control setting. Press the PLAY button on the tape recorder. Press T (for tape). Then press <ENTER>.)

For Disk:
L (in BREAK mode):
The word LOAD and the cursor appear on the screen LOGO: LOAD:__
Enter the appropriate file letter (A-P).

6. Run the procedures to demonstrate that the procedures are now back in the computer's workspace.

Conclude:

Ask several students to talk through the saving and recalling process. If time permits, let several students work through the process. Make a simple chart and post it on the bulletin board or hand out individual reference charts which list the steps necessary to save on cassette or disk.

Students may wish to have their own cassette if using tape recorders. If using disk drives, students can easily share one or two disks. Students should then use their name or initials as their individual file name, or use a combination of name and date or name and number.

Examples: "BOB.3
"MARY.J
"BRIAN.S.4

Add the new primitives to the Primitive Commands wall chart.

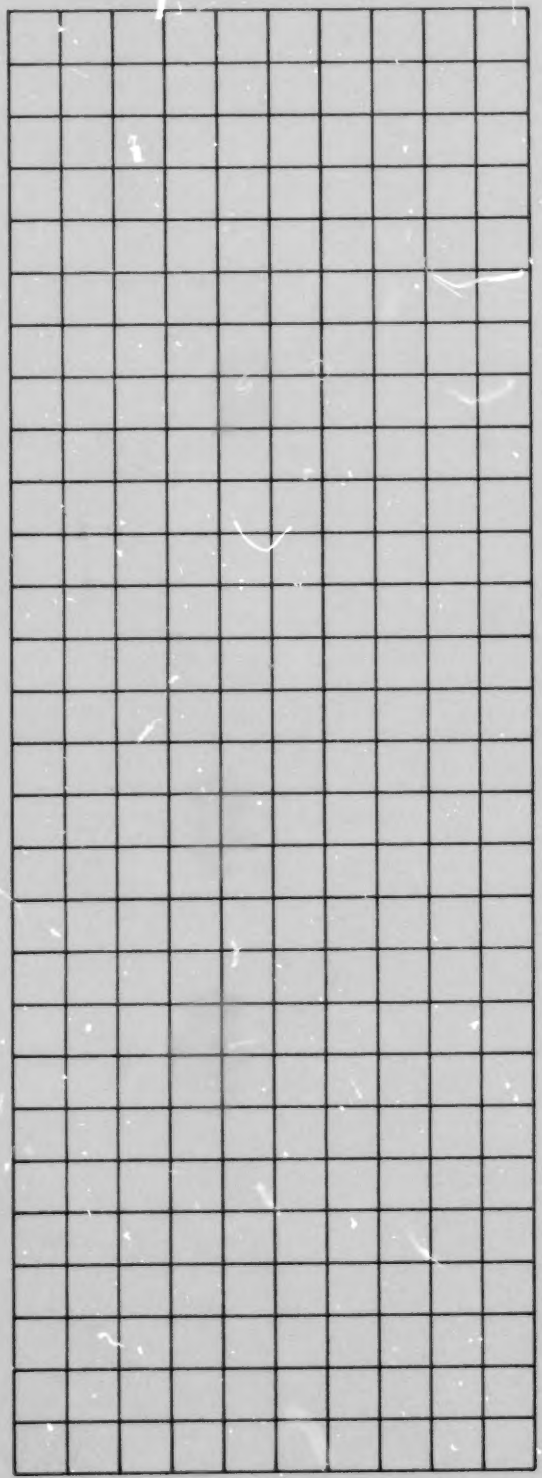
ASSIGNMENT:

Have students sign up for computer time. They should practice saving and recalling a procedure. Have students complete the Problem Sheet.

Session 13
Activity Sheet

Name _____

Date _____



© ICCE 1984. Reproduction for classroom use permitted.

Session 13
Problem Sheet
APPLE, MIT

Name _____
Date _____

Do (1) or (2), or make up a coded message.

1) Decode this Turtle talk:

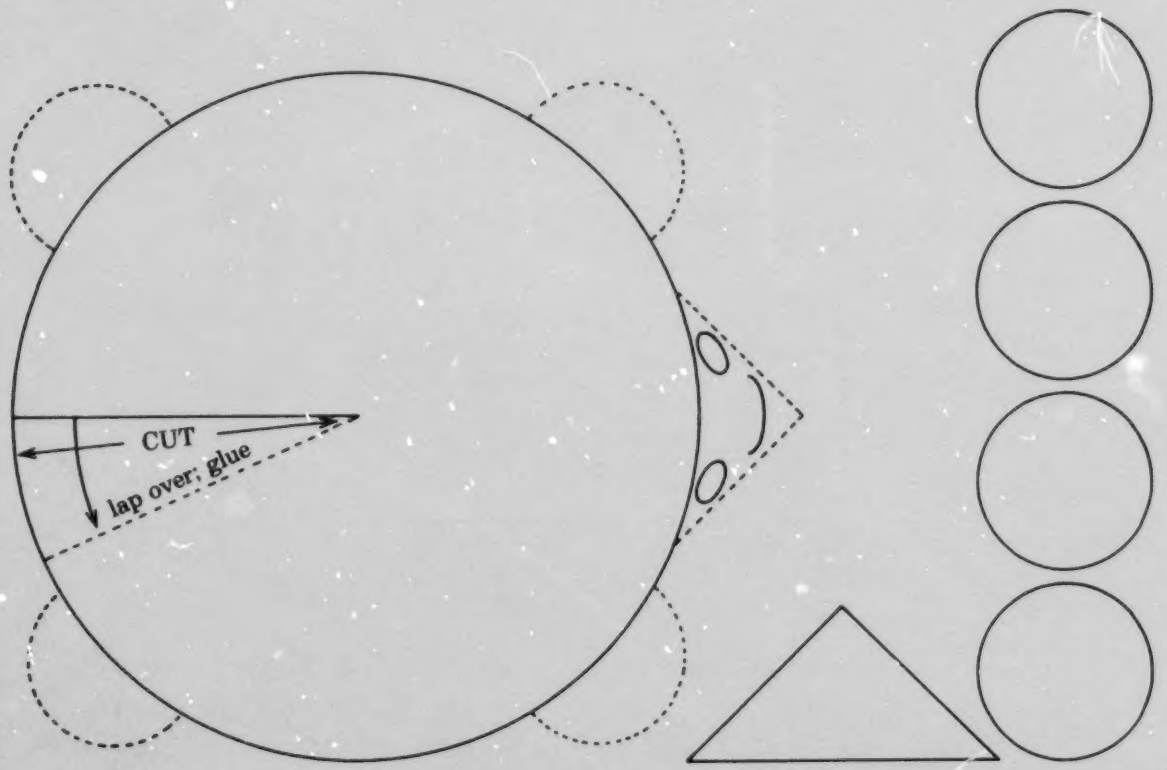
1001 110 11011 11001 1111 10101 11011 1100 1001 1011 101 11011 10100 1111

11 1111 100 101 11011 10100 10010 11001

110 10101 1100 1100 10011 11 10010 101 101 1110 11011 1 1110 100

10011 10000 1100 1001 10100 10011 11 10010 101 101 1110

2) Cut and Color (The Logo Turtle will just go bonkers when it sees all the Turtles!!!)



Head & feet: Cut, color and glue under –see dotted lines.

© ICCE 1984. Reproduction for classroom use permitted.

Session 13
Problem Sheet
ATARI

Name _____
Date _____

Do (1) or (2), or make up a coded message.

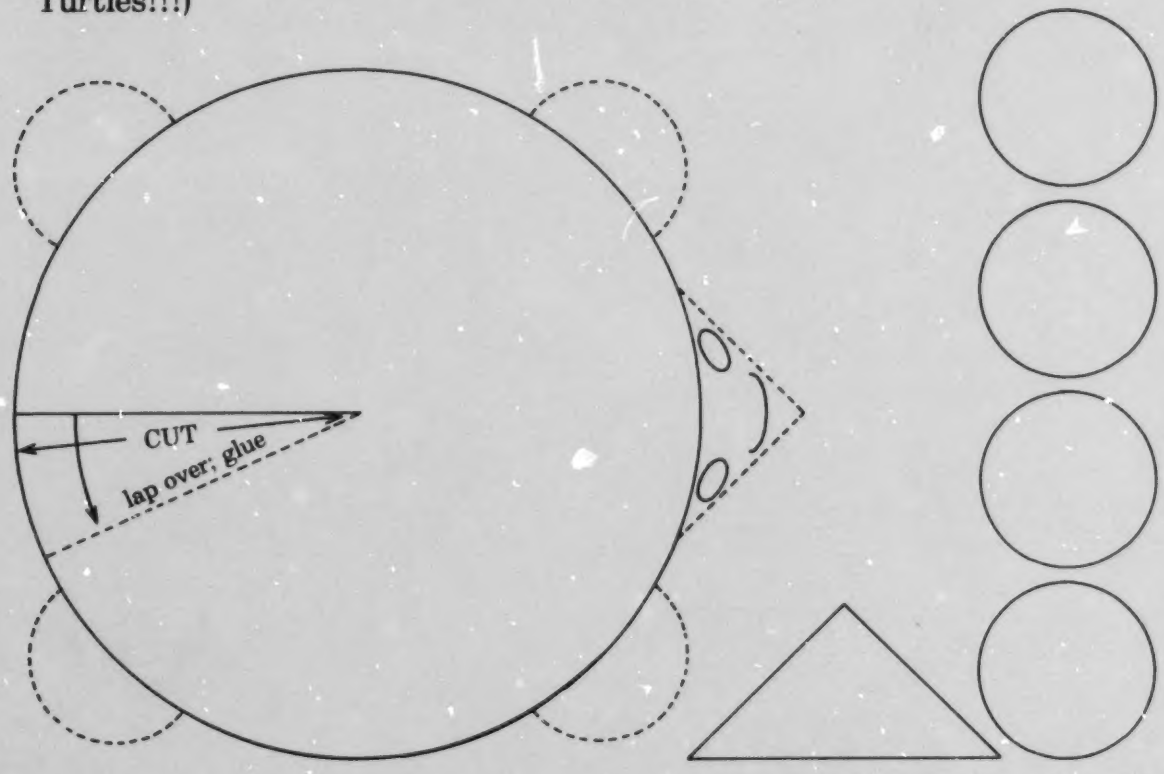
1) Decode this Turtle talk:

1001	110	11011	11001	1111	10101	11011	1100	1001	1011	101	11011	10100	1111

11	1111	100	101	11011	10100	10010	11001	11011	110	10011

1	1110	100	11011	10011	10011

2) Cut and Color (The Logo Turtle will just go bonkers when it sees all the Turtles!!!)



Head & feet: Cut, color and glue under—see dotted lines.

© ICCE 1984. Reproduction for classroom use permitted.

Session 13
Problem Sheet
TI

Name _____
Date _____

Do (1) or (2), or make up a coded message.

1) Decode this Turtle talk:

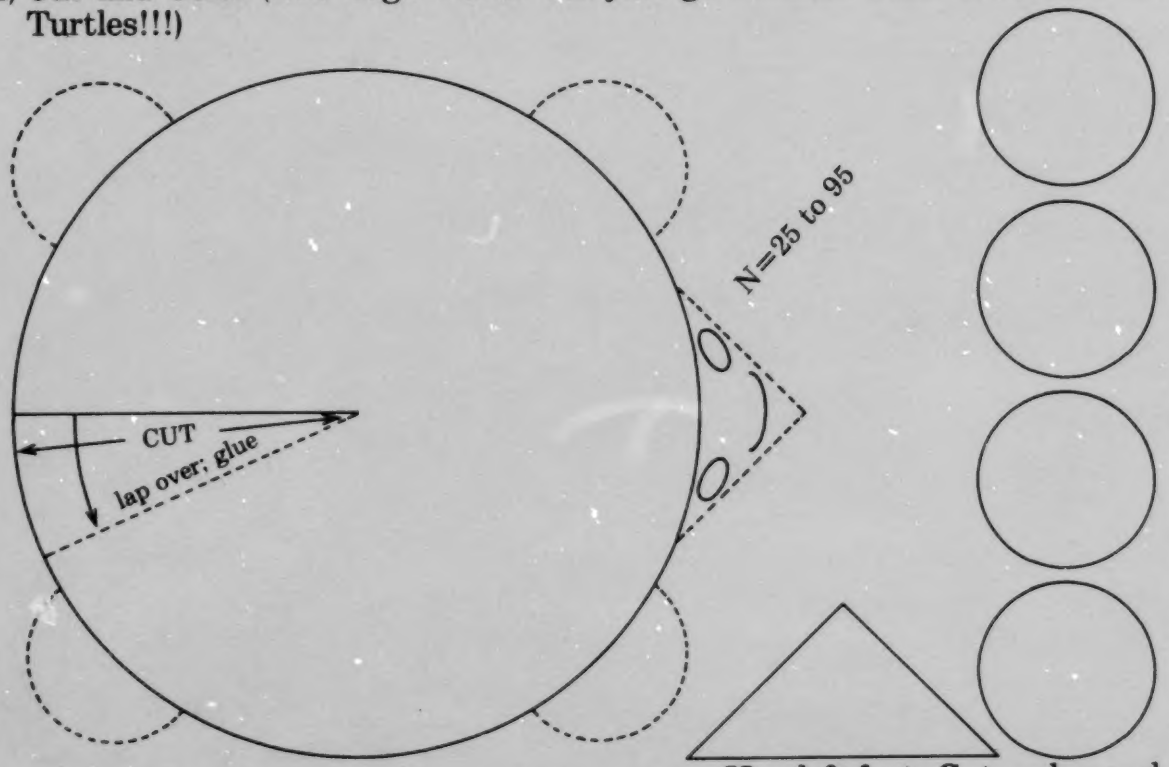
1001 110 11011 11001 1111 10101 11011 1100 1001 1011 101 11011 10100 1111

11 1111 100 101 11011 10100 10010 11001 11011 1110 1111 10100 10101 10010 —

10100 1100 101 11011 1 1110 100 11011 10000 10101 10100 10100 1001 1100 101

1110 11011 11 11011 10010 (N=32 to 95, C=0 to 31, R=0 to 23)

2) Cut and Color (The Logo Turtle will just go bonkers when it sees all the Turtles!!!)



Head & feet: Cut, color and glue under—see dotted lines.

© ICCE 1984. Reproduction for classroom use permitted.

Session 13
Problem Sheet
TRS

Name _____
Date _____

Do (1) or (2), or make up a coded message.

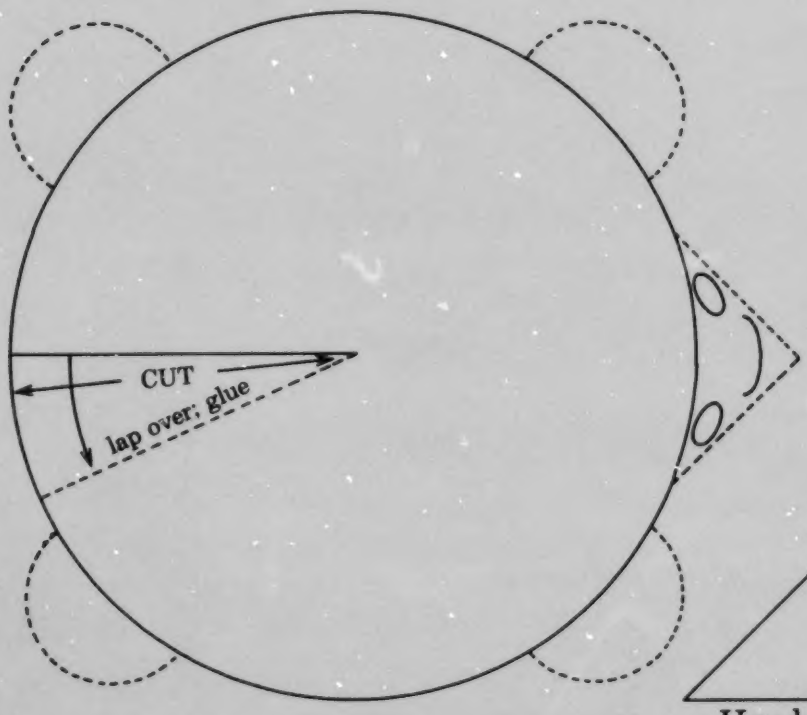
1) Decode this Turtle talk:

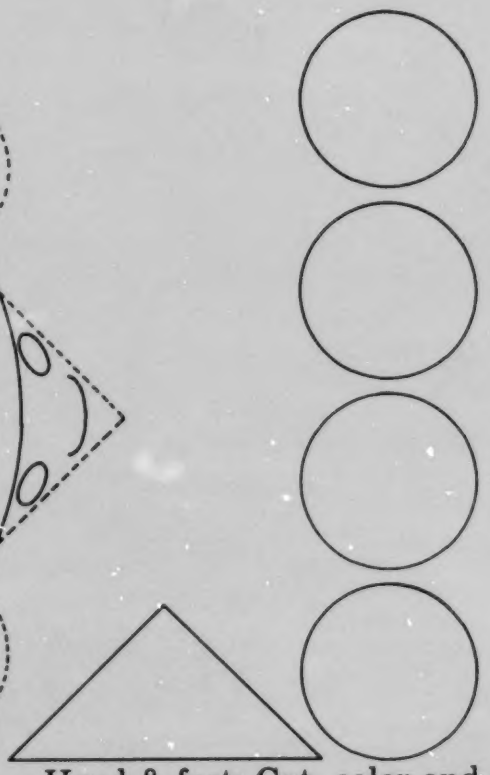
1001 110 11011 11001 1111 10101 11011 1100 1001 1011 101 11011 10100 1111

11 1111 100 101 11011 10100 10010 11001 11011 10011 1100 1111 10111 11011 1110

(N = 0 to 127).

2) Cut and Color (The Logo Turtle will just go bonkers when it sees all the Turtles!!!)





Head & feet: Cut, color and glue under—see dotted lines.

© ICCE 1984. Reproduction for classroom use permitted.

Session 14

*"Turtle geometry is learnable because it is syntonic. And it is an aid to learning other things because it encourages the conscious, deliberate use of problem-solving and mathetic strategies. Mathematician George Polya has argued that general methods for solving problems should be taught. Some of the strategies used in Turtle geometry are special cases of Polya's suggestions. For example, Polya recommends that whenever we approach a problem we should run through a mental checklist of heuristic questions such as: Can this problem be subdivided into simpler problems? Can this problem be related to a problem I already know how to solve?"**

Seymour Papert

Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVES: To introduce students to problem solving in Logo, to introduce the primitive for repetition, and to reinforce working with angle measurements.

CONCEPTS/SKILLS: Problem solving.

FACILITIES/MATERIALS: Overhead or chalkboard; student copies of the Problem Sheet.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)
HIDETURTLE (HT)	HT	HIDETURTLE (HT)	HIDETURTLE (HT)	HIDETURTLE (HT)
SHOWTURTLE (ST)	ST	SHOWTURTLE (ST)	SHOWTURTLE (ST)	SHOWTURTLE (ST)
FULLSCREEN	FS	FULLSCREEN		
SPLITSCREEN	SS	SPLITSCREEN		
			NOTURTLE	
			PUTTILE (PT)	
				SLOW
REPEAT	REPEAT	REPEAT	REPEAT	REPEAT

Defining and Editing Procedures, Managing Workspace and Files

TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E
SAVE	SAVE	SAVE	SAVE	S
LOAD	LOAD	READ	RECALL	L
ERALL	ERALL	ERASE ALL		

*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

COMMENTS:

One of the main goals in education is to help students increase and improve their problem-solving skills. Problem solving is important in every academic discipline.

There are two common approaches to helping students become better problem solvers. The first is to immerse students in a rich problem-solving environment—one with lots of interesting, challenging problems and opportunity for feedback. The second is to explicitly teach ideas of problem solving. There are many books on the general ideas of problem solving and entire college-level courses are taught on this subject.

Logo can be used to help create a very rich problem-solving environment. That is why Logo receives so much praise and support from educators. These educators feel that the Logo problem-solving environment, when backed up by appropriate instructional materials and skilled teachers, can help students to become better problem solvers.

Books on problem solving tend to stress a four-step approach.

1. Understand the problem.
2. Decide on a strategy, or plan.
3. Carry out the plan.
4. Check to see if the problem has been solved.

Such a four-step approach can be discussed with young students. However, memorizing the steps does not automatically improve one's problem-solving skills. What is important is to practice these ideas in a wide variety of problem areas and to internalize these general ideas.

Books on problem solving contain many specific techniques that are applicable to a wide variety of problems. You are probably familiar with techniques such as guess and check, break a problem into smaller problems, look for patterns, and look for similar but simpler examples. All of these techniques can be illustrated and practiced in a Logo environment.

This session introduces the REPEAT primitive command. It is a very useful command in attacking problems that contain repeated patterns. For example, a pentagon can be drawn by five repetitions of FD 40 RT 72. A figure that looks much like a circle can be drawn by 60 repetitions of FD 2 RT 6. Suppose that SQUARE is a procedure to draw a small square. Then 20 repetitions of SQUARE FD 5 RT 18 will produce a nice visual effect.

ACTIVITIES:

Part 1

Patterns are everywhere. Have your whole class work together to make a list of repetitive patterns in your classroom. They may be found in floor or ceiling tiles. rugs, wallpaper, desk and chair arrangement, shelves, windows, etc.

Part 2

Introduce the REPEAT primitive command in Logo. The TRS Logo requires what is to be repeated to be enclosed in curved brackets, as in

REPEAT 4 (FD 50 RT 90) .

The other four versions of Logo covered in this book require square brackets:

REPEAT 4 [FD 50 RT 90]

Help your students to figure out that the above repetitions draw a square.

Ask your students to figure out a way to use REPEAT to write a shorter version of the following procedure.

```
TO HEXAGON
FD 30
RT 60
FD 30
RT 60
FD 30
RT 60
FD 30
RT 60
FD 30
RT 60
FD 30
RT 60
END
```

Part 3

Show your students the following procedure.

```
TO DASH
FD 2
PU
FD 2
PD
END
```

Ask them to determine the figure produced by twenty repetitions of DASH.

ASSIGNMENT:

Have students complete as much of the Problem Sheet as they can before using the computer. Have students sign up for computer time to finish their Problem Sheet.

Session 14 **Name** _____

Problem Sheet **Date** _____

1. Make a list of repeated patterns that you have seen in places outside of your classroom.

2. A student wants to make an octagon (like a STOP sign) by eight repetitions of—

   FD 30 RT _____ .

   Figure out what RT angle is needed and have Turtle draw an octagon. Next, draw the figure first using an angle that is twice as large, then three times as large. Which of the three patterns do you like best?

3. Be sure you have a procedure defined for SQUARE in your workspace. Enter the following sequence of commands using the REPEAT primitive with several different numbers.

   SQUARE RT 36

   Which number of repetitions completes the pattern? _____

4. Write a procedure to have Turtle draw a square with dashed sides.

5. Write a procedure using the REPEAT command to solve ONE of the following:

6. Challenger:
   Can you get Turtle to make this line? Write your procedure on the back of this sheet.

© ICCE 1984. Reproduction for classroom use permitted.

Session 15

*"It is possible to build a large intellectual system without ever making a step that cannot be comprehended. And building with a hierarchical structure makes it possible to grasp the system as a whole. . . ."**

Seymour Papert
Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVES: To teach students to use procedures (subprocedures) within a procedure.

CONCEPTS/SKILLS: Subprocedures.

FACILITIES/MATERIALS: Gym or hallway; paper if you prefer that students not write on the floor; overhead or chalkboard. For each student: 4x6 cards preprinted with TO VOLUNTARY VERTICAL DISPLACEMENT; 4x6 cards used in a previous lesson (TO GET READY, TO ROTATE, TO PALC); water-base, wide-tip pens in several colors for marking on floor or paper. For each pair of students: A set of four blank green 4x6 cards; large protractor; metric ruler; display copy of the Color Chart (see Charts section); and student copies of the two Problem Sheets and Practice Sheet.

PREPARATION: A master for the new procedure (TO VOLUNTARY VERTICAL DISPLACEMENT) is provided to duplicate on tagboard or construction paper and then cut into 4x6 cards. These procedure cards will be used again in the next session. Be sure to use the card for your version of Logo.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)
HIDETURTLE (HT)	HT	HIDETURTLE (HT)	HIDETURTLE (HT)	HIDETURTLE (HT)
SHOWTURTLE (ST)	ST	SHOWTURTLE (ST)	SHOWTURTLE (ST)	SHOWTURTLE (ST)
FULLSCREEN	FS	FULLSCREEN		
SPLITSCREEN	SS	SPLITSCREEN		
			NOTURTLE	
			PUTTILE (PT)	
				SLOW
REPEAT	REPEAT	REPEAT	REPEAT	REPEAT

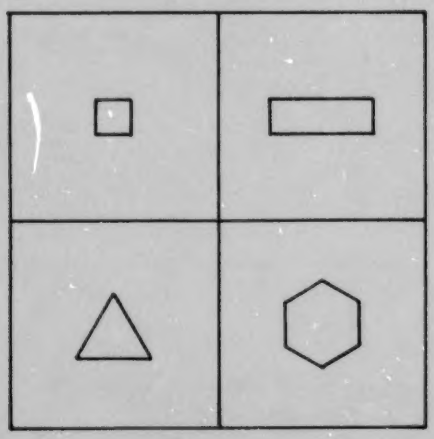
*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

Defining and Editing Procedures, Managing Workspace and Files

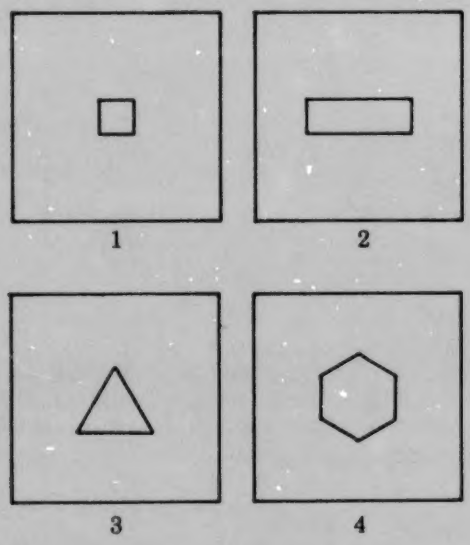
TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E
SAVE	SAVE	SAVE	SAVE	S
LOAD	LOAD	READ	RECALL	L
ERALL	ERALL	ERASE ALL		

COMMENTS:

One of the most important ideas in problem solving is breaking a problem into a collection of smaller problems. Suppose the original problem is to direct Turtle to produce the following figure.



One way to break this into smaller problems is to view it as follows:

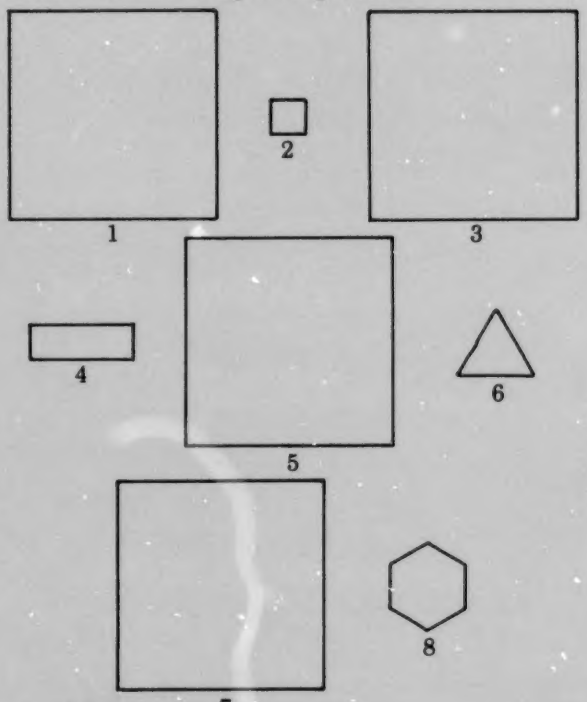


The figures 1, 2, 3, 4 are to be drawn to solve the original problem. The required Logo procedure might have the following format:

```
TO SOLVE ORIGINAL PROBLEM
POSITION TURTLE FOR FIGURE 1
DRAW FIGURE 1
POSITION TURTLE FOR FIGURE 2
DRAW FIGURE 2
POSITION TURTLE FOR FIGURE 3
DRAW FIGURE 3
POSITION TURTLE FOR FIGURE 4
DRAW FIGURE 4
END
```

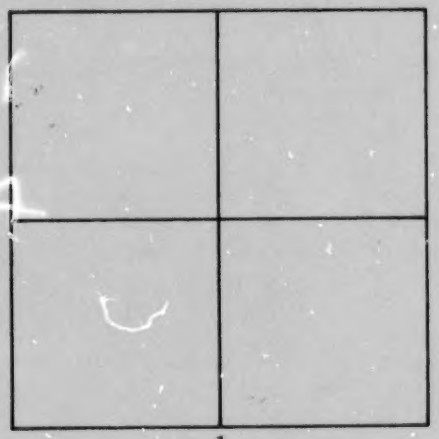
Notice that not only do we have the four problems of drawing figures 1, 2, 3, 4, but we also have the problems of properly positioning Turtle before each figure is to be drawn.

Of course, each of the figures 1, 2, 3, 4 consists of two pieces. Thus, the original problem might be broken down into eight subproblems.

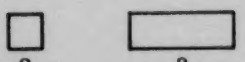


Here we notice that the figures 1, 3, 5, 7 are identical and can each be drawn by a single procedure.

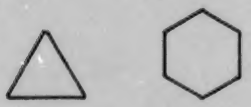
There is no single "right way" to break a problem into smaller problems. For example, the original problem might be viewed as consisting of the following five figures:



1

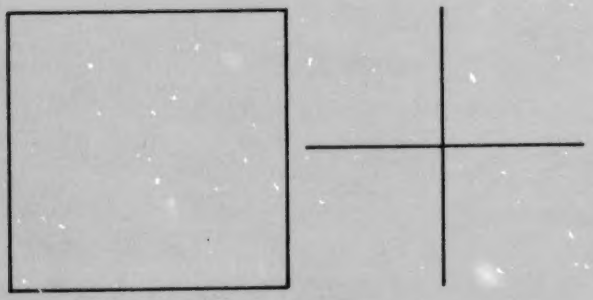


2 3



4 5

Then instead of thinking of Figure 1 as consisting of four small squares, you might think of it as consisting of a large square and a pair of crossed lines.

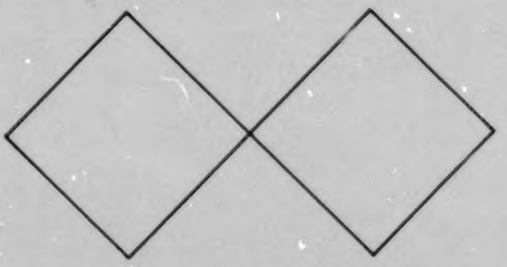


There are many other ways to break the original problem into smaller problems. Generally, some ways to do this will be easier to program, easier to understand or "more elegant" than others. Mathematicians and programmers use the term "elegant" in this context, although they do not precisely define the term.

The main purpose of this session is to give students more time to practice using Logo to solve problems. Your students have been exposed to a very large number of Logo primitives and problem-solving ideas. Continual review and practice are necessary if students are to develop a functional knowledge of this material.

The major new idea introduced in this session is that a procedure may make use of other procedures. Perhaps many of your students have already discovered this fact.

Begin the lesson by reviewing the REPEAT command and playing Turtle. You might want to do this using a Logo procedure to draw the following figure:



With this you could illustrate several distinct ways to solve the same problem.

Also review the primitives for changing background color and pen color. Finally, review PU and PD. You might illustrate by examining the task of drawing the following figure:

ACTIVITY:

Display and recall procedures: GET READY, ROTATE and PALC. Pass out the new card for VOLUNTARY VERTICAL DISPLACEMENT. Have students spread the cards out on their desks and quickly review them.

Explain that they can make a procedure that uses other procedures. Show on overhead:

```
TO SEQUI
GET READY
ROTATE
PALC
VOLUNTARY VERTICAL DISPLACEMENT
PALC
END
```

Tell students that at each step they look at something they already know how to do and follow those directions. Then they return to the procedure for the next set of instructions.

Lead the class through TO SEQUI.

Write "sub" on the overhead or chalkboard. Explain that "sub" means under or below. Show students the five words listed below on the overhead or chalkboard. Discuss their meanings. Some students might locate the words in the dictionary, or perhaps students can add others.

*Sub*marine . . .	Underwater
*Sub*jects	Under the rule or power of another.
*Sub*scribe	To sign name under or at the end of a document (to order a magazine).
*Sub*stitute . . .	Works under the direction or in place of another (substitute teacher).
Sub zero	Below zero.

Call students' attention back to the overhead and to the procedures and ask them what they think the procedures might be called (subprocedures).

```
TO SEQUI
GET READY
ROTATE
PALC                              Each of these
VOLUNTARY VERTICAL                is a
  DISPLACEMENT                    subprocedure.
PALC
END
```

Conclude: "Procedures that are used to define a different procedure are called ___?___" (subprocedures.

Repeat: "Procedures used inside of other procedures are called ___?___" (subprocedures)

Subprocedures are listed **under** the procedure being defined.

Write the following procedure on the overhead, and ask students to name the subprocedures. (SQUARE and TRIANGLE).

```
TO FLAGS
FD 6
SQUARE
FD 20
TRIANGLE
END
```

ASSIGNMENT:

Tell students they are to work with a friend. Pass out a set of four blank 4x6 cards to each pair. Tell students they are to write three procedures: one for a square, one for a triangle, and one for a pentagon or hexagon. Each of these procedures must begin with a primitive to set Turtle's pen color so that when the shapes are drawn, each will be in a different color.

Examples:

```
APPLE: TO SQUARE
       SETPC 2
       REPEAT 4 [ FD 20 RT 90 ]
       END

ATARI: TO SQUARE
       SETPC 0 23
       REPEAT 4 [ FD 20 RT 90 ]
       END

  MIT: TO SQUARE
       PENCOLOR 2
       REPEAT 4 [ FD 20 RT 90 ]
       END

   TI: TO SQUARE
       SETCOLOR :GREEN
       REPEAT 4 [ FD 20 RT 90 ]
       END

  TRS: TO SQUARE
       COLORSET 0
       PENCOLOR 1
       REPEAT 4 ( FD 20 RT 90 )
       END
```

Have students use the fourth card to name a procedure which uses each of the other three procedures as subprocedures. Tell students to keep the procedure simple. The numbers they choose for distances must be small enough for all procedures to be drawn on the gym or hallway floor and large enough to show on the screen when entered into the computer. (For example, FD 2 cm can be drawn on the floor but will not show up well on the video display. Allow students to figure out their own "scale.")

Have students work in pairs in the gym or hallway. Tell them to label a spot called HOME. Then use a meter stick and a large protractor. Play Turtle and draw on the floor the procedure which uses the subprocedures. Each subprocedure will be drawn in a different color.

An alternative to drawing procedures on the floor is to roll out large strips of butcher paper. By drawing on the paper, the products can be displayed on walls.

COMPUTER ASSIGNMENT:

Have students enter their procedures in the computer. Do they look the same?

Have students complete Problem Sheets I and II and enter their work in the computer.

Session 15
Master for TO VOLUNTARY VERTICAL DISPLACEMENT
APPLE, ATARI, MIT, TI

```
TO VOLUNTARY VERTICAL DISPLACEMENT
STAND BY DESK
HANDS AT SIDES
KEEP FEET TOGETHER
REPEAT 3 [ RAISE BOTH FEET ABOUT 20 CM FROM
  FLOOR AND QUICKLY REPLACE ]
END
```

Session 15
Master for TO VOLUNTARY VERTICAL DISPLACEMENT
TRS

```
TO VOLUNTARY VERTICAL DISPLACEMENT
STAND BY DESK
HANDS AT SIDES
KEEP FEET TOGETHER
REPEAT 3 ( RAISE BOTH FEET ABOUT 20 CM FROM
  FLOOR AND QUICKLY REPLACE )
END
```


© ICCE 1984. Reproduction for classroom use permitted.

Session 15 **Name** _____

Problem Sheet I **Date** _____

Write a procedure that makes an attractive screen display by including two regular polygons which use the following primitives at least once:

FD	PD
BK	REPEAT
RT	Command for clearing the screen
LT	Command for changing the pen color
HOME	Command for changing the background color
PU	

TO _____

© ICCE 1984. Reproduction for classroom use permitted.

Session 15
Problem Sheet II
APPLE, ATARI, MIT

Name _____
Date _____

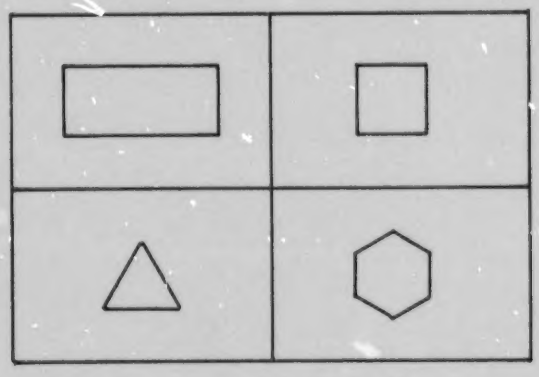
This screen display can be drawn using the BOX and FRAME procedures.

```
TO BOX
FD 75
RT 90
FD 75
RT 90
FD 75
RT 90
FD 75
RT 90
END
```

```
TO FRAME
BOX
RT 90
BOX
RT 90
BOX
RT 90
BOX
RT 90
END
```

1. Make use of the REPEAT command to write shorter versions of BOX and FRAME.

2. Write a procedure which uses FRAME as the second line and then draws the screen display shown below. The shapes do not need to be exactly the same size as in the picture, nor does each shape have to be centered exactly in its window frame. Use the back of this sheet if necessary.



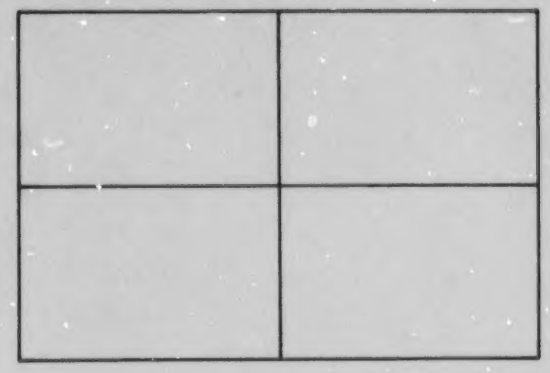
© ICCE 1984. Reproduction for classroom use permitted.

Session 15
Problem Sheet II
TI

Name _____

Date _____

This screen display can be drawn using the BOX and FRAME procedures.

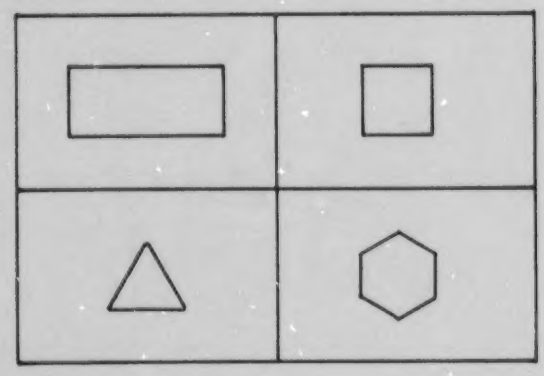


```
TO BOX
FD 45
RT 90
FD 45
RT 90
FD 45
RT 90
FD 45
RT 90
END
```

```
TO FRAME
BOX
RT 90
BOX
RT 90
BOX
RT 90
BOX
RT 90
END
```

1. Make use of the REPEAT command to write shorter versions of BOX and FRAME.

2. Write a procedure which uses FRAME as the second line and then draws the screen display shown below. The shapes do not need to be exactly the same size as in the picture, nor does each shape have to be centered exactly in its window frame. Use the back of this sheet if necessary.

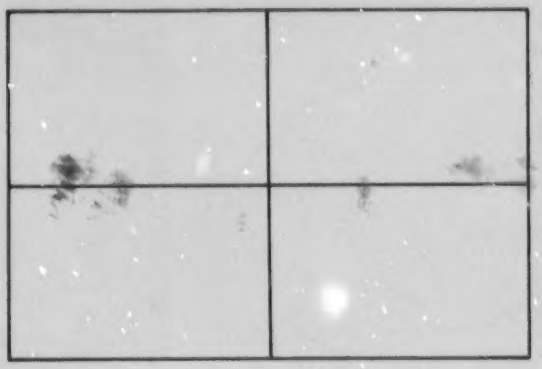


© ICCE 1984. Reproduction for classroom use permitted.

Session 15
Problem Sheet II
TRS

Name _____
Date _____

This screen display can be drawn using the BOX and FRAME procedures.

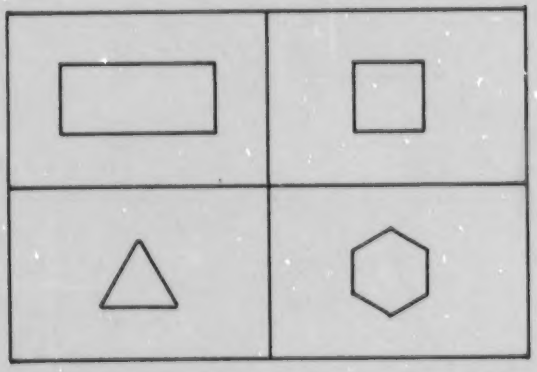


```
TO BOX
FD 60
RT 90
FD 60
RT 90
FD 60
RT 90
FD 60
RT 90
END
```

```
TO FRAME
BOX
RT 90
BOX
RT 90
BOX
RT 90
BOX
RT 90
END
```

1. Make use of the REPEAT command to write shorter versions of BOX and FRAME.

2. Write a procedure which uses FRAME as the second line and then draws the screen display shown below. The shapes do not need to be exactly the same size as in the picture, nor does each shape have to be centered exactly in its window frame. Use the back of this sheet if necessary.



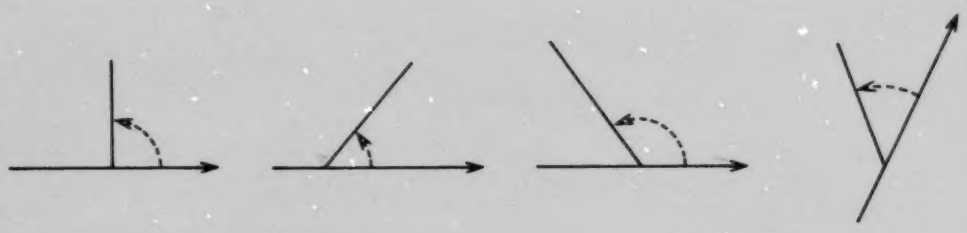
© ICCE 1984. Reproduction for classroom use permitted.

Session 15
Practice Sheet

Name _____
Date _____

Estimating Exercises

First, estimate or guess the size of the angles below. Write down your guess. Then measure the angle with a protractor and write the actual number of degrees.



GUESS: _____ _____ _____ _____
ACTUAL: _____ _____ _____ _____

GUESS: _____ _____ _____
ACTUAL: _____ _____ _____

GUESS: _____ _____
ACTUAL: _____ _____

© ICCE 1984. Reproduction for classroom use permitted.

Session 16

*"Of all ideas I have introduced to children, recursion stands out as the one idea that is particularly able to evoke an excited response. I think this is partly because the idea of going on forever touches on every child's fantasies and partly because recursion itself has roots in popular culture. For example, there is the recursion riddle: If you have two wishes, what is the second? (two more wishes) . . . opening the rich opportunities of playing with infinity . . . puts a child in touch with something of what it is like to be a mathematician."**

Seymour Papert
Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVE: To introduce the concept of recursion.

CONCEPTS/SKILLS: Recursion in Logo.

FACILITIES/MATERIALS: Overhead or chalkboard; dictionary; 1 prepared overhead transparency; 4x6 cards from previous lesson (TO PALC, TO VOLUNTARY VERTICAL DISPLACEMENT, TO ROTATE); preprinted 4x6 cards with TO AAA and TO FUN; computer with Logo cartridge, disk or module; and student copies of Problem Sheet and Challenger Sheet.

PREPARATION: A master is included for the overhead transparency after the ASSIGNMENT section. This information will have to be written on the board if you do not make a transparency. A master for the two new procedures TO AAA and TO FUN is included after the transparency master. This master can be duplicated on construction paper or tagboard and cut into 4x6 cards.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)
HIDETURTLE (HT)	HT	HIDETURTLE (HT)	HIDETURTLE (HT)	HIDETURTLE (HT)
SHOWTURTLE (ST)	ST	SHOWTURTLE (ST)	SHOWTURTLE (ST)	SHOWTURTLE (ST)
FULLSCREEN	FS	FULLSCREEN		
SPLITSCREEN	SS	SPLITSCREEN		
			NOTURTLE	
			PUTTILE (PT)	
				SLOW
REPEAT	REPEAT	REPEAT	REPEAT	REPEAT

*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

124

Defining and Editing Procedures, Managing Workspace and Files

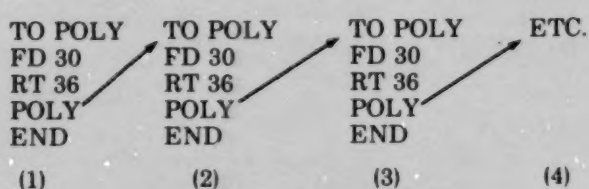
TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E
SAVE	SAVE	SAVE	SAVE	S
LOAD	LOAD	READ	RECALL	L
ERALL	ERALL	ERASE ALL		

COMMENTS:

This session introduces a powerful new idea called **recursion. Recursion** is the ability for a procedure to call a **copy** of itself. Consider the following example.

```
TO POLY
FD 30
RT 36
POLY
END
```

Notice that POLY is part of the definition of POLY. As Logo begins to execute the procedure POLY it will follow the commands in lines 2 and 3. It is then directed to execute the procedure POLY. A diagram for what happens is given below.



Imagine having many copies of the procedure POLY. The computer begins to execute (1), the first copy of POLY. But when it gets to the fourth line, it is directed to the second copy of POLY. When it reaches the fourth line in (2), the computer is directed to the third copy of POLY.

As you can see, this process continues indefinitely. The result is that FD 30 RT 36 is repeated indefinitely. This type of recursion is an easy way to direct Turtle to repeat some task an infinite number of times. Each version of Logo contains provisions for stopping such an infinite recursion and you will want your students to learn these provisions. They are listed later in this session.

With the Logo knowledge your students currently have, recursion is of limited value. For example, compare the previous POLY procedure with POLY1.

```
TO POLY1
REPEAT 10000 [ FD 30 RT 36 ]
END
```

While 10,000 repetitions is not an infinite number, it is probably a satisfyingly large number.

But as your students get further into Logo, they will find recursion is a very powerful aid to problem solving. A number of applications of recursion and how to help students learn recursion are given in the December/January 1983-84 issue of *The Computing Teacher.* At this stage of students' development recursion can be presented as a fun way to make interesting patterns.

ACTIVITY:

Write on the chalkboard or overhead:

RECUR

Discuss meanings. Have someone look up "recur" in the dictionary. One meaning is to occur or happen again. Write this meaning on the board or overhead.

Tell students that today they're going to learn how to use an idea called **recursion** in Logo.

Write the following procedures on the overhead or chalkboard. Wait until the demonstration to add the arrows.

```
TO PIX
RT 90
FD 290
LT 90
SQUARE
FD 30
LT 90
BK 30
TRIANGLE
PU
HOME
PD
HT
END

TO SQUARE
REPEAT 4 [ FD 30 RT 90 ]*
END

TO TRIANGLE
REPEAT 3 [ FD 30 RT 120 ]*
END
```

*TRS—Use parentheses instead of brackets.

Go through the set of directions as you draw the path Turtle would make. (You may want to use the computer in immediate mode, one step at a time, if the screen is easily seen by all.)

Tell students that this is an example of how Logo uses subprocedures. Stress the subprocedure routine by drawing an arrow from the subprocedure SQUARE to the SQUARE card, completing the di-

rections and then returning to the place from which SQUARE was called, to continue the procedure PIX.

Repeat with the subprocedure TRIANGLE.

The concept to develop is that of returning to the place from which the calling procedure left off and continuing with the next command.

Have students stand by their desks.

Pass out the 4x6 cards: PALC, VOLUNTARY VERTICAL DISPLACEMENT, ROTATE and AAA.

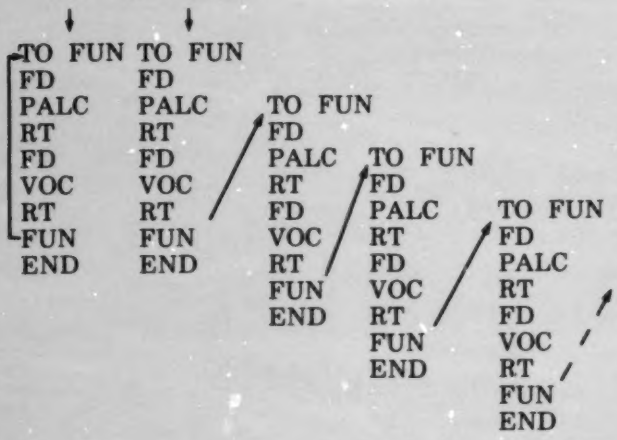
Allow time for all students to successfully complete the set of directions on TO AAA. Stress the following: **When a procedure card is called as a subprocedure, look at that card, do that set of directions, and return to the place in the main procedure** to finish the original set of directions, even if the only command left to finish is END.

Pass out TO FUN.

Slowly go through TO FUN. When the subprocedure FUN is reached, tell students this command calls a subprocedure FUN, to which they must go next. **It does not mean to loop back up to the procedure currently being used.**

Use the transparency or write the following procedures on the overhead or chalkboard. You will need to add arrows to the transparency.

Not this, but this:



Tell students that the computer people who developed Logo compared recursion to the idea of a child who was given two wishes. The second wish is for two more wishes, which makes it keep going on forever.

Tell students that later they will learn ways to stop a recursive procedure within the procedure. For now, have students do FUN and discover they cannot stop, because each time FUN is called, they must go to another copy of the procedure FUN.

To stop an infinitely recursive procedure in Logo from outside the procedure:

APPLE: <CTRL>G
ATARI: <CTRL>G
MIT: <CTRL>G
TI: Hold down <FCTN> and press <BACK>
TRS: <BREAK>

Conclude: Recursive procedures can be written for Turtle. Demonstrate on the computer:

```
TO SQUARE
REPEAT 4 [ FD 40 RT 90 ]*
SQUARE
END
```

```
TO DESIGN
LT 10
HEX
DESIGN
END
```

```
TO HEX
REPEAT 6 [ FD 30 RT 60 ]*
END
```

```
TO OCTOCT
FD 15
OCTA
OCTOCT
END
```

```
TO OCTA
REPEAT 8 [ FD 20 RT 45 ]*
END
```

*TRS—Use parentheses instead of brackets.

After the demonstration, ask students to tell which of the five procedures are recursive and which are not.

ASSIGNMENT:

Have students complete the Problem Sheet. Those who like can do the Challenger Sheet.

Session 16
Transparency

```
TO FUN
FD
PALC
RT
FD
VOC
RT
FUN
END
```

```
TO FUN
FD
PALC
RT
FD
VOC
RT
FUN
END
```

```
TO FUN
FD
PALC
RT
FD
VOC
RT
FUN
END
```

```
TO FUN
FD
PALC
RT
FD
VOC
RT
FUN
END
```

```
TO FUN
FD
PALC
RT
FD
VOC
RT
FUN
END
```


© ICCE 1984. Reproduction for classroom use permitted.

```
TO AAA
FD 2 STEPS
PALC
RT 90
FD 2 STEPS
VOLUNTARY VERTICAL DISPLACEMENT
RT 90
FD 2 STEPS
ROTATE
RT 90
FD 2 STEPS
RT 90
END
```

```
TO FUN
FD 2 STEPS
PALC
RT 180
FD 2 STEPS
VOLUNTARY VERTICAL DISPLACEMENT
RT 180
FUN
END
```


© ICCE 1984. Reproduction for classroom use permitted.

Session 16 **Name** _____

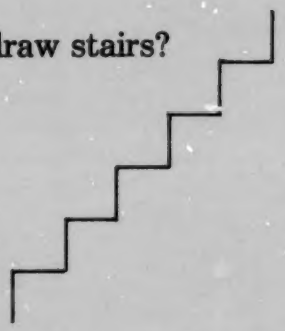
Problem Sheet, Page 1 **Date** _____

The following set of directions will draw a step:

```
FD 15
RT 90
FD 15
LT 90
```

1. Can you write a recursive procedure that will draw stairs?

   TO STAIRS

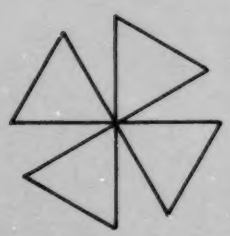


The following procedure will draw a triangle:

```
TO TRI
FD 30
RT 120
FD 30
RT 120
FD 30
RT 120
END
```

2. Rewrite the procedure, making use of a REPEAT to shorten the procedure.

3. Write a recursive procedure that makes use of TRI to make the following screen display.



© ICCE 1984. Reproduction for classroom use permitted.

Name _____

Date _____

Challenger: The procedure PIC is recursive and makes use of STAR to draw a pleasing picture.

```
TO STAR
FD 12
RT 144
FD 12
RT 144
FD 12
RT 144
FD 12
RT 144
FD 12
RT 144
END
```

```
TO PIC
FD 60
STAR
BK 60
RT 25
PIC
END
```

4. Rewrite STAR, making use of REPEAT to shorten the procedure.

5. Type the procedure STAR and PIC into the computer, and then execute PIC. Does the picture Turtle is drawing ever get completed?

6. The RT 25 in procedure PIC controls how much turning occurs between pieces of the picture being drawn. Experiment with values such as RT 30, RT 45, RT 90 and RT 180. Which picture do you like best? Why?

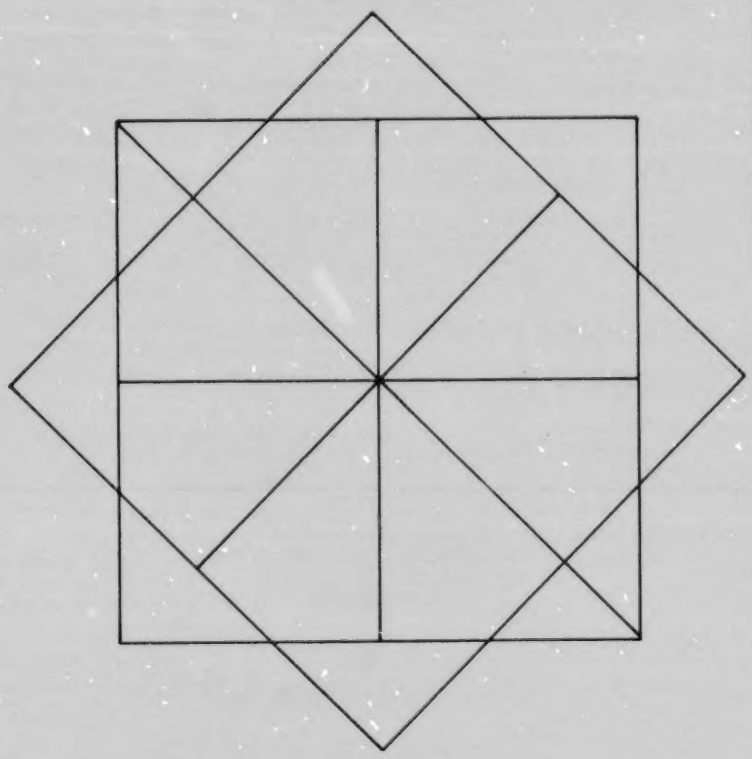
© ICCE 1984. Reproduction for classroom use permitted.

Session 16
Challenger Sheet

Name _____
Date _____

Do you remember Hungry Cat from Session 1 at the beginning of your Logo instructions? The first problem you had to solve was to help Hungry Mouse find the dish of food behind the door and then help Mouse send directions for that Hungry Cat to find the food. The problem was that both Mouse and Cat could only turn LEFT. When Mouse found the food he sent directions to Hungry Cat, but since he didn't really want to share the food with that cat, he sent directions that put Hungry Cat into a recursive path like a dog chasing its tail. Below is the path that Hungry Cat made. Can you write the same directions Mouse sent H.C.? (I really don't know how H.C. ever got out, do you??)

Remember: Only LEFT TURNS!!!



TO H.C.DIZZY

Try making a simple recursive design and copying it on construction paper with markers or crayons. Write the procedure in the corner and display it on the bulletin board.

131

© ICCE 1984. Reproduction for classroom use permitted.

Session 17

"In this chapter we focus on two important mathetic principles. . . . First, relate what is new and to be learned to something you already know. Second, take what is new and make it your own. . . . Piaget has studied the spontaneous learning of children and found both steps at work—the child absorbs the new into the old in a process that Piaget calls **assimilation**, *and the child constructs his knowledge in the course of actively working with it."**

Seymour Papert
Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVES: To allow time on the computer for free exploration and creativity; to allow time for assimilating and internalizing Logo primitives and geometric concepts at the child's individual developmental level.

CONCEPTS/SKILLS: Free exploration for reinforcement of skills and concepts previously introduced.

FACILITIES/MATERIALS: Primitive Commands wall chart; graph paper (see Charts section); student copies of Problem Sheet.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)
HIDETURTLE (HT)	HT	HIDETURTLE (HT)	HIDETURTLE (HT)	HIDETURTLE (HT)
SHOWTURTLE (ST)	ST	SHOWTURTLE (ST)	SHOWTURTLE (ST)	SHOWTURTLE (ST)
FULLSCREEN	FS	FULLSCREEN		
SPLITSCREEN	SS	SPLITSCREEN		
			NOTURTLE	
			PUTTILE (PT)	
				SLOW
REPEAT	REPEAT	REPEAT	REPEAT	REPEAT

*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E
SAVE	SAVE	SAVE	SAVE	S
LOAD	LOAD	READ	RECALL	L
ERALL	ERALL	ERASE ALL		

COMMENTS:

This is a review session. It provides time for students to practice the ideas they have studied in previous sessions.

Encourage your students to plan and carry out an in-depth project. Some ideas for projects and/or how to plan a project are given in the **ACTIVITY** section.

ACTIVITY:

Part 1

Review metric graph paper. Discuss the following carefully with students.

Scale: 1 mm=the length of one Turtle step.

Remind students that there are 10 mm to 1 cm, but this paper shows only 5 parts to 1 cm. Graph paper that shows 10 parts in 1 cm has the lines so close together that the lines are hard to see. Since there are only 5 parts in 1 cm on this paper, the length of one of the 5 parts is the length of 2 Turtle steps and the length of 10 Turtle steps is 1 cm.

Shapes and pictures can be drawn on the graph paper. Mark HOME in the center of the graph paper. If the intended picture is large, students may need to tape two or three sheets of paper together. However, the picture will have to be limited to screen size. After the picture is complete, write procedures to draw the picture on the screen. A few simple models can be displayed so students can readily follow the process.

Approximate screen sizes: (There may be slight differences depending on monitor or TV set used.)

APPLE COMPUTER:
- Home to top of screen=120 Turtle steps
- Home to bottom of screen=119 Turtle steps (Fullscreen)
- Home to bottom of screen=78 Turtle steps (Splitscreen)
- Home to either side of screen=139 Turtle steps

ATARI:
- Home to top of screen=120 Turtle steps
- Home to bottom of screen=119 Turtle steps
- Home to either side of screen=159 Turtle steps

COMMODORE 64:
- Home to top of screen=130 Turtle steps
- Home to bottom of screen=127 Turtle steps
- Home to right side of screen=159 Turtle steps
- Home to left side of screen=160 Turtle steps

TI:
- Home to top of screen=95 Turtle steps
- Home to bottom of screen=46 Turtle steps
- Home to either side of screen=120 Turtle steps

TRS-80 COLOR COMPUTER:
- Home to top of screen=87 Turtle steps
- Home to bottom of screen=64 Turtle steps
- Home to either side of screen=120 Turtle steps

Part 2

Put the following five procedures on the overhead or chalkboard.

```
TO F        TO B
FD 5        BK 5        TO M
END         END         PU
                        FD 5
TO R        TO L        PD
RT 90       LT 90       END
END         END
```

Notice that each of these procedures has a one-letter name that suggests what the procedure does. Thus, F suggests "forward" while L suggests "left turn" and M suggests "move without drawing."

Ask various students in your class to explain in their own words what the procedures do. Then have each student carry out on graph paper the following sequences of procedures.

a. F R F R F R F R
b. F M F M F M F M
c. F F F R F F F R F F F R M F R F R F R F R

This is similar to the single-keystroke type of Logo procedures often used with young children. It is a powerful idea. The five procedures taken together define a language consisting of five 1-letter words. Suggest to your students that they may want to create their own versions of single keystroke Logo.

ASSIGNMENT:

Allow students free exploration time this week. They may write procedures, create pictures, or just talk in the immediate mode to Turtle. It's not necessary to use the graph paper at this time.

Encourage students to ask if they have a special problem or need special directions for something they want to make.

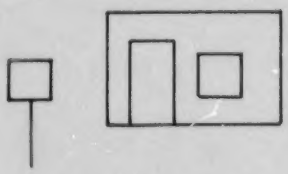
Ideas are suggested in the Problem Sheet.

Session 17
Problem Sheet
APPLE, ATARI, MIT, TI

Name _____
Date _____

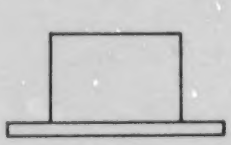
If you choose to use the graph paper to create a picture, here are some suggestions and a sample to use as a model. Using a model is another way to solve a problem.

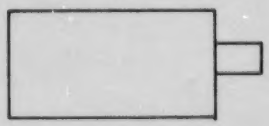
HOUSE

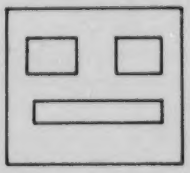


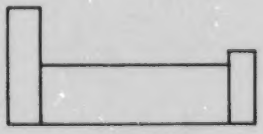
```
TO RECTANGLE
REPEAT 2 [ FD 50 RT 90 FD 80
  RT 90 ]
END

TO SMALLREC
REPEAT 2 [ FD 20 RT 90 FD 40
  RT 90 ]
END

TO SQUARE
REPEAT 4 [ FD 20 RT 90 ]
END

TO HOUSE
RECTANGLE
LT 90
BK 30
SMALLREC PU
BK 10
RT 90
FD 10 PD
SQUARE PU
BK 20
LT 90
FD 80
RT 90 PD
FD 30
LT 90
BK 10
SQUARE
HT
END
```

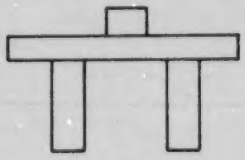
Here are some shapes to draw on the graph paper. The length of 1 small square on the paper = 2 Turtle steps. The length of 1 large square = 10 Turtle steps. Write the procedure using subprocedures and *try it!*

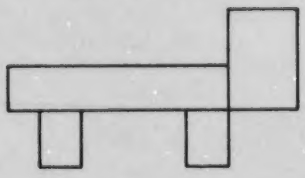


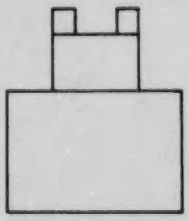












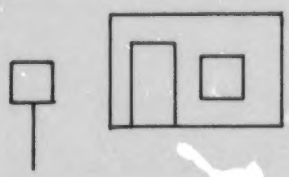
© ICCE 1984. Reproduction for classroom use permitted.

Session 17
Problem Sheet
TRS

Name _____
Date _____

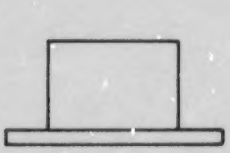
If you choose to use the graph paper to create a picture, here are some suggestions and a sample to use as a model. Using a model is another way to solve a problem.

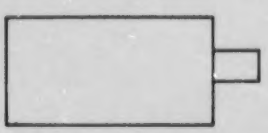
HOUSE

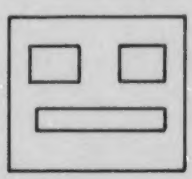


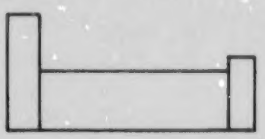
```
TO RECTANGLE
REPEAT 2 ( FD 50 RT 90 FD 80
  RT 90 )
END

TO SMALLREC
REPEAT 2 ( FD 20 RT 90 FD 40
  RT 90 )
END

TO SQUARE
REPEAT 4 ( FD 20 RT 90 )
END

TO HOUSE
RECTANGLE
LT 90
BK 30
SMALLREC PU
BK 10
RT 90
FD 10 PD
SQUARE PU
BK 20
LT 90
FD 80
RT 90 PD
FD 30
LT 90
BK 10
SQUARE
HT
END
```

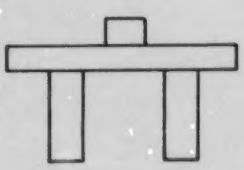
Here are some shapes to draw on the graph paper. The length of 1 small square on the paper = 2 Turtle steps. The length of 1 large square = 10 Turtle steps. Write the procedure using subprocedures and *try it!*

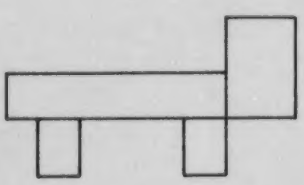


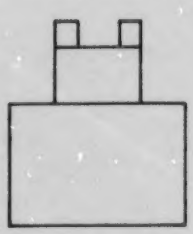












© ICCE 1984. Reproduction for classroom use permitted.

Session 18

*". . . the concept of symbolic naming through a variable, one of the most powerful mathematical ideas ever invented. . . . The name could be a letter, such as X, or it could be a whole word, such as ANGLE or DISTANCE. (One of the minor contributions of the computer culture to mathematics is its habit of using mnemonic words instead of single letters as names for variables.) To put the idea of variable to work, TURTLE TALK allows one to create a 'procedure with an input.'" **

Seymour Papert
Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVE: To introduce students to the use of variables as size input for geometric shapes.

CONCEPTS/SKILLS: Variables and their use in procedures.

FACILITIES/MATERIALS: Overhead or chalkboard; small milk carton covered with construction paper; small box; several tagboard strips approximately 5x15 cm, 4x6 card preprinted with the procedure TO CLAP; slip of paper for each student; and student copies of the Practice Sheet.

PREPARATION: Cut off the top of a small milk carton, cover the outside with construction paper and label it NUM. Label the 5x15 cm tagboard strips with different single-digit numerals written on the top fourth of the front and back. Label the small box VARIABLE VALUES. Number the slips of paper with a number from 3 to 10. Preprint a 4x6 card with the following procedure:

```
TO CLAP :NUMBER
REPEAT :NUMBER [ CLAP YOUR HANDS TOGETHER ONCE ] *
END
```

*TRS—Use parentheses instead of brackets.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)
HIDETURTLE (HT)	HT	HIDETURTLE (HT)	HIDETURTLE (HT)	HIDETURTLE (HT)
SHOWTURTLE (ST)	ST	SHOWTURTLE (ST)	SHOWTURTLE (ST)	SHOWTURTLE (ST)
FULLSCREEN	FS	FULLSCREEN		
SPLITSCREEN	SS	SPLITSCREEN		
			NOTURTLE	
			PUTTILE (PT)	
				SLOW
REPEAT	REPEAT	REPEAT	REPEAT	REPEAT

*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E
SAVE	SAVE	SAVE	SAVE	S
LOAD	LOAD	READ	RECALL	L
ERALL	ERALL	ERASE ALL		

COMMENTS:

This session introduces the idea of a variable in Logo. Your students may have already been exposed to variables in their math studies.

$$6 + \square = 8$$

What number must be placed in the box to make this a correct equation?

The "box" notation of elementary school mathematics is a simplified version of algebraic notation. In an algebra course, the above problem would be written as:

$6 + X = 8$
Solve for X.

The idea of variables in computer programming is a combination of the box from elementary school mathematics with algebraic notation. A variable is memory space (that is, a memory box) that has a name.

The key idea is to distinguish between the name of the memory space and the contents of the memory space. The analogy with a person's mailbox is often used. The person's name is on the mailbox, and that name remains the same day after day. But the contents of the mailbox change from day to day.

ACTIVITY:

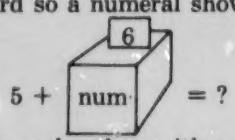
Part 1

When discussing the concept of "variable," you may want to have students look up the word "vary" in the dictionary. Attaching the suffix "able" alters the meaning to become capable of changing.

Work with the class to make a list of things that are capable of changing. Examples include a person's height, weight or age; the weather; the price of a candy bar.

Part 2

Using appropriately large printing, put the following number sentence on the board. Then tape the milk carton on the board over the [num] box. Insert a strip of tagboard so a numeral shows above the box.



Take out the six and replace with a new strip displaying a different numeral. Repeat as students give answers until they develop the idea that [num] can vary.

Part 3

Write on the overhead or chalkboard:

```
TO SQ1
REPEAT 4 [ FD 5 RT 90 ] *
END

TO SQ2
REPEAT 4 [ FD 10 RT 90 ] *
END

TO SQ3
REPEAT 4 [ FD 25 RT 90 ] *
END
```

*TRS—Use parentheses instead of brackets.

Discuss with your students what is the same about these three procedures and what is different about them. All draw a square. But the length of the side of the square varies. Suggest that it would be convenient if Logo made it possible to easily vary the length of the side in a square without having to rewrite the procedure.

Part 4

Put the following procedure on the overhead or chalkboard.

```
TO SQUARE :LENGTH
REPEAT 4 [ FD :LENGTH RT 90 ] *
END
```

*TRS—Use parentheses instead of brackets.

Tell your students that :LENGTH means the value of LENGTH in Logo. Lead a discussion that covers the following points:

- LENGTH is a variable.
- :LENGTH is the value of that variable.
- This one procedure can draw squares of varying sizes.

More specifically, if the previous procedure has been typed into a Logo system, then SQUARE 5 will draw a square whose sides have length 5; SQUARE 10 will produce a square with sides of length 10; SQUARE 25 will produce a square with sides of length 25.

The idea of using a variable in the title line of a procedure definition is new. But the idea of providing a value to a Logo command, as in FD 20 or RT 60, is now familiar to students.

Part 5

Use the 4x6 card preprinted with the procedure CLAP. Place all of the numbered slips of paper in the box labeled VARIABLE VALUES.

Have all students stand up. Give one student the procedure card and the box of slips of paper. The student is to draw a slip of paper from the box and read its number. That number becomes the value of the variable NUMBER in the CLAP procedure. The student is to execute CLAP, hand the procedure card and box of papers to a person who is standing up, and then sit down. This process continues until every person (including teacher) has had a chance to CLAP.

Optional Activity

This activity creates a nice visual effect when viewed from a distance (much like viewing a Turtle drawing in progress on a **large** screen!) and illustrates the process of creating different-sized squares by varying the input to the procedure.

You need a medium for drawing trails—tempera or yarn, for example. Different colors might be used. The drawing area should be a fairly large space where one group of students creates the drawing while other students watch the process from an elevated distance such as bleachers or a second or third story window. A large sheet of tagboard with SQUARE :NUM written on it and large sheets of paper to write the input values on completes the simulation of the screen.

Students creating the drawing take turns assigning different values to NUM, printing the number in large print on a sheet of paper (which is placed over :NUM on the tagboard) and displaying it. Then one student walks the procedure while another marks the trail. The next pair of students assigns a different value to NUM and follows the same steps in adding their square to the drawing.

ASSIGNMENT:

Have students complete the Practice Sheet.

Session 18
Practice Sheet
APPLE

Name _____

Date _____

Write a procedure using a variable and the REPEAT primitive. Try running several together on the video display. Try using different colors. Write your favorite here:

Here are some samples to help you get started:

```
TO HEX :NUM
SETPC 5 SETBG 6
REPEAT 6 [ FD :NUM RT 60 ]
LT 150
HEX :NUM
END
```

```
TO OCT :NUM
SETPC 2 SETBG 4
HT
REPEAT 8 [ FD :NUM RT 45 ]
RT 40
OCT :NUM
END
```

```
TO MYPIC :NUM :SIZE
SETPC 4 SETBG 3
HEXA :NUM
TRI :SIZE
LT 30
TRI :SIZE
MYPIC :NUM :SIZE
END
```

```
TO HEXA :NUM
REPEAT 6 [ FD :NUM RT 60 ]
END

TO TRI :SIZE
REPEAT 3 [ FD :SIZE LT 120 ]
END
```

JUST FOR FUN

- Draw your design on drawing paper and enlarge it using the opaque projector. Color and display it.
- Draw your design on manila drawing paper. Put a piece of wax paper over the design. Lay lightweight yarn over the lines. Cover with another piece of wax paper. Iron lightly. Trim. Hang it in the window.
- Draw your design on manila drawing paper. Cover with white tissue paper so you can still see the design. Lightly glue colored tissue strips over the design or dot the white tissue with felt pens. Hang it in the window.

© ICCE 1984. Reproduction for classroom use permitted.

Session 18
Practice Sheet
ATARI

Name _____

Date _____

Write a procedure using a variable and the REPEAT primitive. Try running several together on the video display. Try using different colors. Write your favorite here:

Here are some samples to help you get started:

```
TO HEX :NUM
SETPC 0 72 SETBG 3
REPEAT 6 [ FD :NUM RT 60 ]
LT 150
HEX :NUM
END
```

```
TO OCT :NUM
SETPC 0 99 SETBG 28
HT
REPEAT 8 [ FD :NUM RT 45 ]
RT 40
OCT :NUM
END
```

```
TO MYPIC :NUM :SIZE
SETPC 0 18 SETBG 40
HEXA :NUM
TRI :SIZE
LT 30
TRI :SIZE
MYPIC :NUM :SIZE
END
```

```
TO HEXA :NUM
REPEAT 6 [ FD :NUM RT 60 ]
END

TO TRI :SIZE
REPEAT 3 [ FD :SIZE LT 120 ]
END
```

JUST FOR FUN

- Draw your design on drawing paper and enlarge it using the opaque projector. Color and display it.
- Draw your design on manila drawing paper. Put a piece of wax paper over the design. Lay lightweight yarn over the lines. Cover with another piece of wax paper. Iron lightly. Trim. Hang it in the window.
- Draw your design on manila drawing paper. Cover with white tissue paper so you can still see the design. Lightly glue colored tissue strips over the design or dot the white tissue with felt pens. Hang it in the window.

© ICCE 1984. Reproduction for classroom use permitted.

Session 18
Practice Sheet
MIT

Name _____

Date _____

Write a procedure using a variable and the REPEAT primitive. Try running several together on the video display. Try using different colors. Write your favorite here:

Here are some samples to help you get started:

```
TO HEX :NUM
PC 5 BG 6
REPEAT 6 [ FD :NUM RT 60 ]
LT 150
HEX :NUM
END
```

```
TO OCT :NUM
PC 2 BG 4
HT
REPEAT 8 [ FD :NUM RT 45 ]
RT 40
OCT :NUM
END
```

```
TO MYPIC :NUM :SIZE
PC 4 BG 3
HEXA :NUM
TRI :SIZE
LT 30
TRI :SIZE
MYPIC :NUM :SIZE
END
```

```
TO HEXA :NUM
REPEAT 6 [ FD :NUM RT 60 ]
END

TO TRI :SIZE
REPEAT 3 [ FD :SIZE LT 120 ]
END
```

JUST FOR FUN

- Draw your design on drawing paper and enlarge it using the opaque projector. Color and display it.
- Draw your design on manila drawing paper. Put a piece of wax paper over the design. Lay lightweight yarn over the lines. Cover with another piece of wax paper. Iron lightly. Trim. Hang it in the window.
- Draw your design on manila drawing paper. Cover with white tissue paper so you can still see the design. Lightly glue colored tissue strips over the design or dot the white tissue with felt pens. Hang it in the window.

© ICCE 1984. Reproduction for classroom use permitted.

Session 18
Practice Sheet
TI

Name _____

Date _____

Write a procedure using a variable and the REPEAT primitive. Try running several together on the video display. Try using different colors. Write your favorite here:

Here are some samples to help you get started:

```
TO HEX :NUM
SC 3 CB 15
REPEAT 6 [ FD :NUM RT 60 ]
LT 150
HEX :NUM
END
```

```
TO OCT :NUM
SC 4 CB 9
HT
REPEAT 8 [ FD :NUM RT 45 ]
RT 40
OCT :NUM
END
```

```
TO MYPIC :NUM :SIZE
SC 4 CB 15
HEXA :NUM
TRI :SIZE
LT 30
TRI :SIZE
MYPIC :NUM :SIZE
END
```

```
TO HEXA :NUM
REPEAT 6 [ FD :NUM RT 60 ]
END

TO TRI :SIZE
REPEAT 3 [ FD :SIZE LT 120 ]
END
```

JUST FOR FUN

- Draw your design on drawing paper and enlarge it using the opaque projector. Color and display it.
- Draw your design on manila drawing paper. Put a piece of wax paper over the design. Lay lightweight yarn over the lines. Cover with another piece of wax paper. Iron lightly. Trim. Hang it in the window.
- Draw your design on manila drawing paper. Cover with white tissue paper so you can still see the design. Lightly glue colored tissue strips over the design or dot the white tissue with felt pens. Hang it in the window.

© ICCE 1984. Reproduction for classroom use permitted.

Session 18
Practice Sheet
TRS

Name _____

Date _____

Write a procedure using a variable and the REPEAT primitive. Try running several together on the video display. Try using different colors. Write your favorite here:

Here are some samples to help you get started:

```
TO OCT :NUM
COLORSET 1 PC 1
HT
REPEAT 8 ( FD :NUM RT 45 )
RT 40
OCT :NUM
END
```

Note: *Before* you type HEX (to try the procedure in the RUN mode), type BG 1.

```
TO HEX :NUM
COLORSET 1 PC 0
REPEAT 6 ( FD :NUM RT 60 )
LT 150
HEX :NUM
END
```

Note: *Before* you type MYPIC n n (to try the procedure in the RUN mode), type BG 2.

```
TO MYPIC :NUM :SIZE
COLORSET 0 PC 1
HEXA :NUM
TRI :SIZE
LT 30
TRI :SIZE
MYPIC :NUM :SIZE
END

TO HEXA :NUM
REPEAT 6 ( FD :NUM RT 60 )
END

TO TRI :SIZE
REPEAT 3 ( FD :SIZE LT 120 )
END
```

JUST FOR FUN

- Draw your design on drawing paper and enlarge it using the opaque projector. Color and display it.
- Draw your design on manila drawing paper. Put a piece of wax paper over the design. Lay lightweight yarn over the lines. Cover with another piece of wax paper. Iron lightly. Trim. Hang it in the window.
- Draw your design on manila drawing paper. Cover with white tissue paper so you can still see the design. Lightly glue colored tissue strips over the design or dot the white tissue with felt pens. Hang it in the window.

© ICCE 1984. Reproduction for classroom use permitted.

Session 19

*"How can I make the Turtle draw a circle? The instructor in a LOGO environment does not provide answers to such questions but rather introduces the child to a method for solving not only this problem but a large class of others as well. This method is summed up in the phrase "play Turtle." The child is encouraged to move his or her body as the Turtle on the screen must move in order to make the desired pattern. For the child who wanted to make a circle, moving in a circle might lead to a description such as: 'When you walk in a circle you take a little step forward and you turn a little. And you keep doing it.' . . . Another child, perhaps less experienced in simple programming and in the heuristics of 'playing Turtle,' might need help. But the help would not consist primarily of teaching the child how to program the Turtle circle, but rather of teaching the child a method, a heuristic procedure."**

Seymour Papert

Mindstorms: Children, Computers, and Powerful Ideas

OBJECTIVE: To introduce/reinforce creating circles of various sizes on the computer.

CONCEPTS/SKILLS: Reinforcing self as a point of reference and using a table as a strategy for problem solving.

FACILITIES/MATERIALS: Overhead or chalkboard; transparency for Session 19; graph paper for each student; student copies of Problem Sheet and Challenger Sheet.

PREPARATION: A master for a transparency of the chart used in Session 19 is included after the ASSIGNMENT section.

PRIMITIVE CHART:

Turtle Graphics, Screen Commands

APPLE	ATARI	MIT	TI	TRS
CLEARSCREEN (CS)	CS	DRAW	TELL TURTLE	R
FD	FD	FD	FD	FD
BK	BK	BK	BK	BK
RT	RT	RT	RT	RT
LT	LT	LT	LT	LT
HOME	HOME	HOME	HOME	HOME
CLEAN	CLEAN	CS	CS	CLEAR
PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)	PENUP (PU)
PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)	PENDOWN (PD)
PENERASE (PE)	PE	PENCOLOR (PC)	PENERASE (PE)	PENCOLOR (PC)
				COLORSET
SETPC	SETPC	PENCOLOR (PC)	SETCOLOR (SC)	PENCOLOR (PC)
SETBG	SETBG	BACKGROUND (BG)	COLORBACKGROUND (CB)	BACKGROUND (BG)
HIDETURTLE (HT)	HT	HIDETURTLE (HT)	HIDETURTLE (HT)	HIDETURTLE (HT)
SHOWTURTLE (ST)	ST	SHOWTURTLE (ST)	SHOWTURTLE (ST)	SHOWTURTLE (ST)
FULLSCREEN	FS	FULLSCREEN		
SPLITSCREEN	SS	SPLITSCREEN		
			NOTURTLE	
			PUTTILE (PT)	
				SLOW
REPEAT	REPEAT	REPEAT	REPEAT	REPEAT

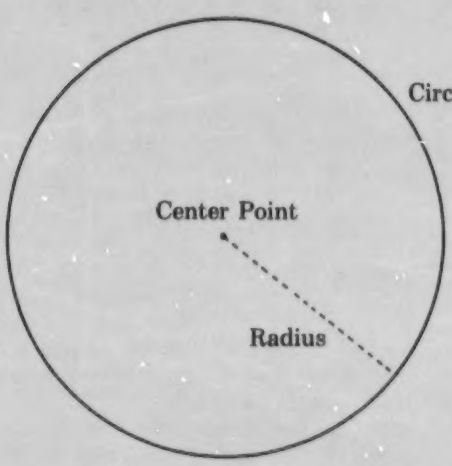
*From *Mindstorms: Children, Computers, and Powerful Ideas* by Seymour Papert. © 1980 by Basic Books Inc., Publishers. By permission of the publisher.

Defining and Editing Procedures, Managing Workspace and Files

TO	TO	TO	TO	TO
END	END	END	END	END
POTS	POTS	POTS	PP	
ERASE (ER)	ERASE (ER)	ERASE	ERASE	
EDIT	EDIT	EDIT	TO	E
SAVE	SAVE	SAVE	SAVE	S
LOAD	LOAD	READ	RECALL	L
ERALL	ERALL	ERASE ALL		

COMMENTS:

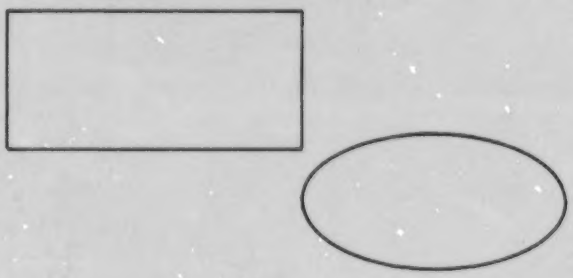
This session discusses circles. In the Euclidean plane, a circle is the set of all points that are equidistant from a given point. The given point is the center of the circle and the distance is the radius.



In Logo, a standard approach to studying and drawing circles is to consider the regular polygons. Students look at a progression of regular polygons with a steadily increasing number of sides. Certainly an equilateral triangle or a square doesn't look much like a circle. But a hexagon or an octagon are more circle-like in appearance.

Do not mislead your students by telling them that a 36-sided regular polygon or a 360-sided regular polygon is a circle. Both are still polygons drawn by a sequence of straight line segments. But in looking at a sequence of regular polygons with more and more sides, something mathematically significant may occur to some of your students. They may form the intuition (to use Piaget's term) that a circle is an infinite-sided polygon with turns approaching zero! Encourage this type of intuitive thinking.

As you know, the screen for most Logo systems can display only a finite number of distinct points. These points are close enough together so that Turtle appears to draw line segments. On some Logo systems the length of a segment made by FD 10 is not the same in the horizontal and vertical directions. In such systems a square will turn out looking like a (non-square) rectangle and a many-sided regular polygon will look like a (non-circle) ellipse.



If this occurs on your Logo system, consult your Logo manual for details of how to deal with the problem. If the problem cannot easily be corrected, you should discuss the problem with your students.

ACTIVITY:

Part 1

Review how to make regular polygons by having students walk an equilateral triangle, a square, and a regular pentagon. Review that the total turning needed to make a regular polygon is 360°. Review how to find the number of degrees to turn at each vertex to form a regular polygon (360°/number of sides).

Part 2

After carrying out Part 1, ask students how to walk a circle.

Discuss the necessity of having some forward movement combined with some turning. Lead them to realize that since there are 360 degrees in one rotation, the amount of turning must total 360 degrees. Let students try walking a circle while you write all suggestions on the overhead or chalkboard. Now, develop with your students a procedure for making circles. Begin by writing on the overhead or chalkboard the problem-solving steps used previously:

1. **Understand the problem.**
2. **Decide on a strategy, or plan.**
3. **Carry out the plan.**
4. **Check to see if the problem has been solved.**

If the problem is understood, tell students that today they are going to use a table or chart as a strategy. Use the overhead or chalkboard to display the following table:

TO ROTATE

REPEAT (Number of Sides)	FORWARD	RIGHT	TOTAL TURNING

Begin with the square—4 sides. Add a side each time and have students construct the chart. Use calculators for division.

360/4
360/5

The completed table should look like this:

TO ROTATE

REPEAT	FORWARD	RIGHT	TOTAL TURNING
4	10	90	360
5	10	72	360
6	10	60	360
7	10	51*	360
8	10	45	360
9	10	40	360
10	10	36	360
12	10	30	360
15	10	24	360
18	10	20	360
20	10	18	360

*Rounded.

Complete Step 4 of the problem-solving method by checking the results in the computer. For example: REPEAT 20 [FD 10 RT 18]*. This can be done as a class activity or during individual computer time.

Note: Some Logo systems such as the TRS and the TI round decimal fractions to the nearest whole number. For example, 360/7 rounds to 51. Attempts to make a 7-sided regular polygon with turns of 51° will produce a figure that doesn't quite close upon itself.

*TRS uses parentheses instead of brackets.

Part 3

The number of sides a regular polygon needs to look like a circle varies with the Logo system being used. In the example below we assume 36 suffices. If that doesn't work well for your machine, you will want to use a different number.

Put the following procedure on the overhead or chalkboard.

```
TO POLYCIR :SIZE
REPEAT 36 [ FD :SIZE RT 10 ] *
END
```

Make sure that your students understand that POLYCIR can be used to draw a circle-like figure and that the actual size of the figure depends on the value of the variable SIZE. Thus, POLY 5 produces a small circle while POLY 20 produces a larger circle.

Part 4

In previous sessions we have looked at recursion and at procedures making use of variables. Of course, these can be combined. Use an overhead or chalkboard to display the following procedure.

```
TO SQUARE :SIZE
REPEAT 4 [ FD :SIZE RT 90 ] *
SQUARE :SIZE+5
END
```

Help your students to figure out what Turtle will do as it attempts to execute SQUARE 10. After completing a square with sides of length 10, Turtle encounters SQUARE 15. So Turtle draws a square with sides of length 15 and then encounters SQUARE 20, etc.

ASSIGNMENT:

Have students use a sheet of graph paper to create a picture using straight lines and polygons. They should try to include different-sized polygons that appear as circles on the screen.

Have students complete the Problem Sheet for Session 18.

A Challenger Sheet is included for interested students.

Session 19
Transparency

TO ROTATE

REPEAT	FORWARD	RIGHT	TOTAL TURNING

© ICCE 1984. Reproduction for classroom use permitted.

Session 19 **Name** _____

Problem Sheet **Date** _____

APPLE, ATARI, MIT, TI

Fill in this table. Can you see the pattern? Try some during your computer time.

REPEAT	FD	RT
360	1	1
180	2	2
120	3	3
36	10	10

Fill in this table. Find the pattern. Make notes or draw some circles on the back of this sheet and label so you know which numbers to use for small circles and larger circles.

REPEAT	FD	RT
360	1	1
180	1	2
120	1	3
36	1	10

Just for fun, try these:

```
TO GROW :N
FD :N RT 90
GROW :N+2
END

TO CIR
REPEAT 36 [ FD 1 RT 10 ]
END
```

```
TO TWOCIR
RT 90 PU
BK 90 PD
CIR
FD 20
CIR
PU
FD 10
PD
TWOCIR
END
```

```
TO PLA :N
FD 10 RT :N
PLA :N+30
END
```

© ICCE 1984. Reproduction for classroom use permitted.

Session 19 **Name** _____

Problem Sheet **Date** _____

TRS

Fill in this table. Can you see the pattern? Try some during your computer time.

REPEAT	FD	RT
360	1	1
180	2	2
120	3	3
36	10	10

Fill in this table. Find the pattern. Make notes or draw some circles on the back of this sheet and label so you know which numbers to use for small circles and larger circles.

REPEAT	FD	RT
120	3	3
90	3	4
36	3	10

Just for fun, try these:

```
TO GROW :N
FD :N RT 90
GROW ( :N+2 )
END
```

```
TO PLA :N
FD 10 RT :N
PLA ( :N+30 )
END
```

© ICCE 1984. Reproduction for classroom use permitted.

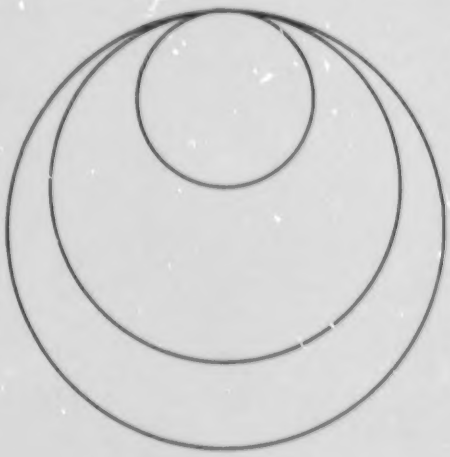
Session 19
Challenger Sheet

Name _____
Date _____

Can you make these designs with Turtle? Remember to write down your directions! You may want to use these designs in a procedure you write later.

1)

2)



3)

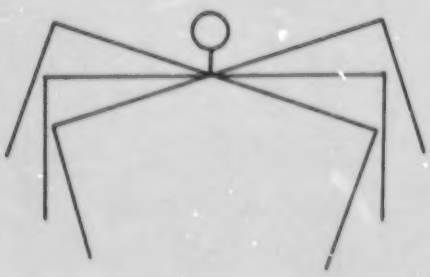
4)

5)

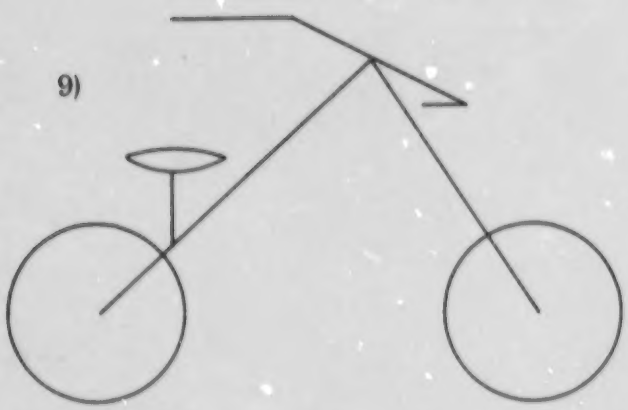
6)

7)

8)



9)



150

© ICCE 1984. Reproduction for classroom use permitted.

Session 20

COMMENTS:

Logo is a programming language and these sessions have introduced your students to only a small part of the language. Indeed, Logo is a large and powerful language—it has been used in the first course for college computer science majors.

But Logo is much more than a programming language. Logo is a vehicle for an educational philosophy that can lead to significant improvements in the educational system.

It is a careful blend of the concrete (we can watch Turtle move) and the abstract (symbolic notation). Thus, Logo can help students bridge the gap between the Piagetian "concrete" and "formal operations" levels.

Logo is an excellent vehicle for studying problem solving. Of course, it is easy enough to have students memorize four standard steps in problem solving.

1. Understand the problem.
2. Decide on a strategy,or plan.
3. Carry out the plan.
4. Check to see if the problem has been solved.

But these steps have no inherent meaning for most students.

Logo is an environment in which one can practice problem solving. For many students the task of directing Turtle to produce a specified pattern is intrinsically interesting. Moreover, students are continually encouraged to make up their own problems. (Decide on a pattern you would like Turtle to draw, and then tell Turtle how to draw it.)

It is easy to say,"2. Decide on a strategy, or plan," but generally this is quite difficult to do. There are strategies such as break the problem into smaller problems and look for patterns. These strategies are quite useful in many Logo problems, so a student gets practice in using them.

In Logo, "3. Carry out the plan" involves writing procedures and giving commands to Turtle. Here students are practicing algorithmic thinking and the careful, precise communication of a set of directions.

Perhaps the greatest value of Logo as an instructional tool occurs in "4. Check to see if the problem has been solved." In Logo the most frequent outcome is that the result is partly right and partly not right. This means that over and over again students reach step 4 and must return to a previous step. Logo allows students to provide their own feedback (they can "see" that Turtle has not drawn the desired pattern). Providing one's own feedback and understanding the role of debugging are two of the most important ideas in problem solving!

This last session is for review and summary. Your goals are to help students realize that they have made a tremendous amount of progress and to make them feel good about themselves. There are many ways to do this; what you do must fit your students, your school setting and you.

We recommend a Logo party in which parents, grandparents, other relatives, friends, etc. are invited. A Logo party involves decorating the room, displaying some work of each student and demonstrations (each child demonstrates to his/her guests and other people). But the party is also a celebration, a time for fun and relaxation.

Of course, the party should not signal the end of your students' work with Logo. Obtain a file-card box (4" by 6" is preferred) and a supply of cards. Use these cards to create a a large collection of problems for your students. Store them in the card file box and label it THE LOGO PROBLEM BOX. Make the box accessible to your students. Some possibilities for problems are given after the ASSIGNMENT section, but you will want to make up some for yourself and make use of ideas from magazines, books and your students.

ASSIGNMENT

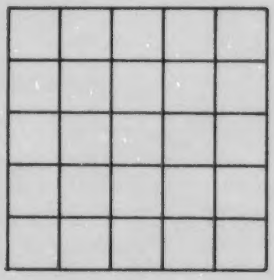
1. Go to THE LOGO PROBLEM BOX, select a problem, and solve it.
2. Make up a problem you feel should be in THE LOGO PROBLEM BOX. Solve it. With your teacher's help, prepare the problem on a card to be placed in THE LOGO PROBLEM BOX.

Session 20

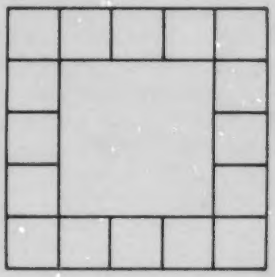
Task Cards for the Logo Problem Box, Page 1

Think about both A. and B. before you begin. How are the two problems related?

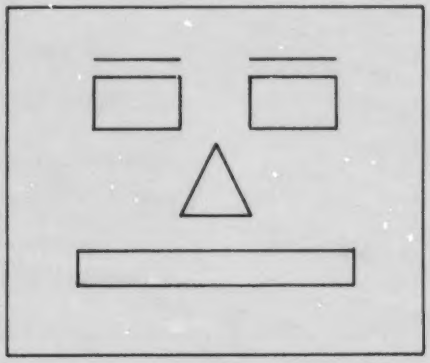
A. Direct Turtle to draw the following pattern.



B. Direct Turtle to draw the following pattern.



Direct Turtle to draw a clown face or a robot face. It can look like the one given below, or you can make up your own pattern.

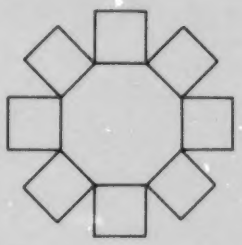


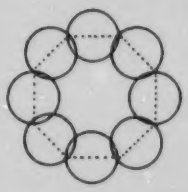
© ICCE 1984. Reproduction for classroom use permitted.

Session 20

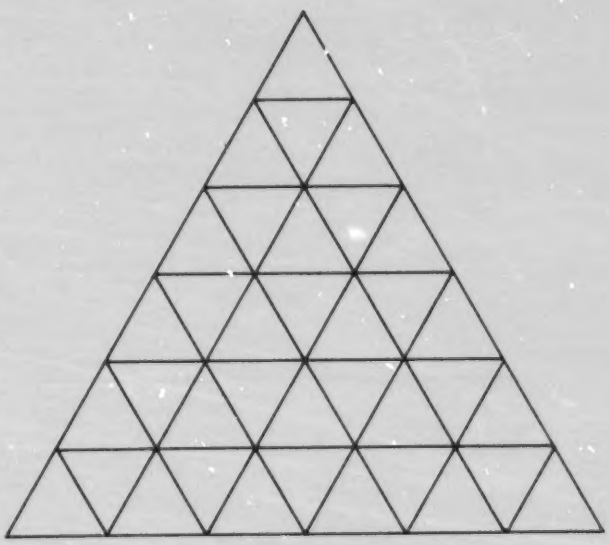
Task Cards for the Logo Problem Box, Page 2

Direct Turtle to draw a large circle made up from small squares or from small circles. Hint: The "circle" you draw is a many-sided regular polygon. Use a small square or a small circle in place of each of the sides of the polygon.





Direct Turtle to draw a large triangle made up of smaller triangles.

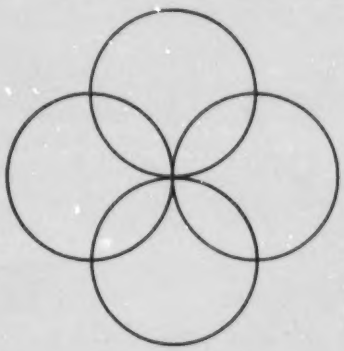


© ICCE 1984. Reproduction for classroom use permitted.

Session 20
Task Cards for the Logo Problem Box, Page 3

Make use of a procedure that draws a circle.

A. Direct Turtle to draw the following figure.



Hint: Draw a circle, do a RT 90. draw a circle, do a RT 90, etc.

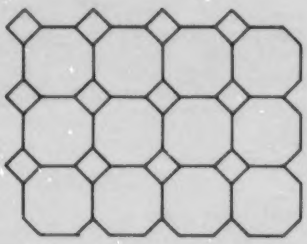
B. Experiment with drawing a sequence of circles with a turn other than RT 90 between circles.

© ICCE 1984. Reproduction for classroom use permitted.

Session 20
Repeated Pattern Tasks, Page 1

On the next set of task cards, students write procedures to fill the screen with repeated designs. Before you add these cards to the Logo Problem Box, you may want to walk students through this example.

A simple pattern can be repeated to produce an attractive design on the screen. This design is a repetition of a square and an octagon.

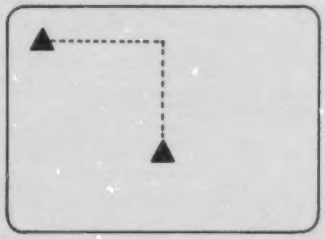


To write a procedure that will tell Turtle how to draw this design on the screen, look at the simple pattern that is repeated. To draw the pattern, Turtle will need procedures for a SQUARE and an OCTAGON.

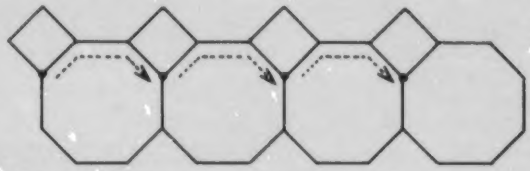
```
TO SQUARE
REPEAT 4 [ FD 20 RT 90 ]
END

TO OCTAGON
REPEAT 4 [ FD 20 RT 90 ]
END

TO PATTERN
LT 45
SQUARE
RT 90
OCTAGON
END
```

To draw a row of the repeated patterns, Turtle will need a procedure to move it to a beginning location on the screen. One way to do this is to use the primitives FD, RT, LT, PU and PD. Remember to start Turtle at HOME heading straight up.

```
TO SETUP
PU HOME
FD 90
LT 90
FD 100
RT 90
PD
END
```



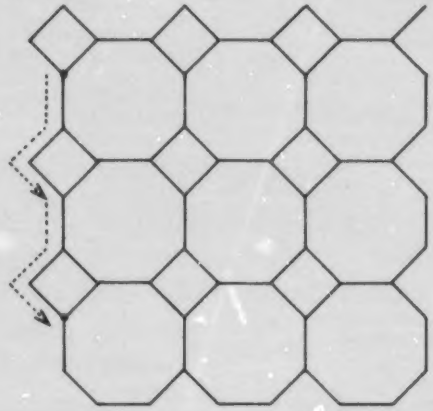
To draw a row of the repeated patterns, Turtle will need a procedure that will move it from one starting pattern-location to the next starting pattern-location.



One way to do this is shown below.

```
TO MOVE.OVER
FD 20 RT 45
FD 20 RT 45
FD 20 LT 135
END

TO ROW
REPEAT 3 [ PATTERN MOVE.OVER ]
PATTERN
END

TO ROW1
SETUP
ROW
END
```

To draw another row of the repeated patterns, Turtle will need a procedure to move from the end of a row to the starting position of the next row. One way to do this would be to move Turtle back to the original starting position by using the SETUP procedure. Then move Turtle down to the beginning of the second row with a procedure called MOVE.DOWN.



Session 20
Repeated Pattern Tasks, Page 2

```
TO MOVE.DOWN
PU
BK 20
RT 45
BK 20
LT 90
BK 20
RT 45
PD
END
```

```
TO ROW2
SETUP
MOVE.DOWN
ROW
END
```

To draw the third row, again begin with SETUP. By using the procedure MOVE.DOWN Turtle will be in position for the second row. Now use MOVE.DOWN again so Turtle is in position for the third row.

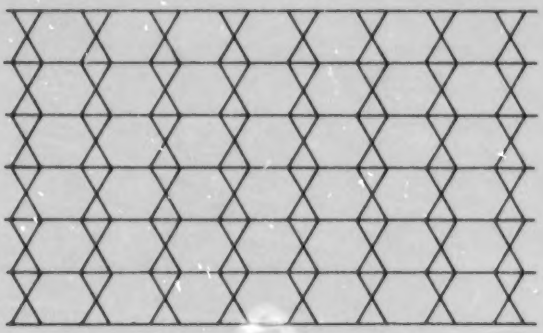
```
TO ROW3
SETUP
MOVE.DOWN
MOVE.DOWN
ROW
END
```

Now, combine all the procedures into a super procedure to make the design on the screen so that you have at least four repetitions of the pattern in each of three rows.

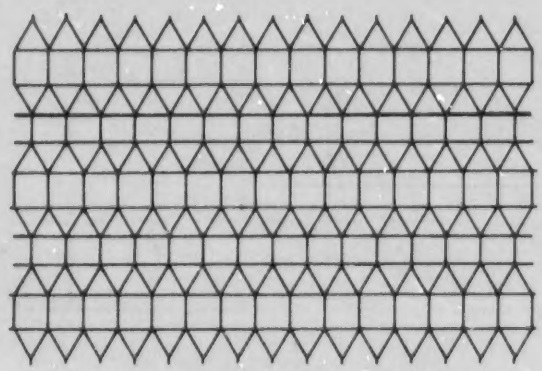
```
TO DESIGN
ROW1
ROW2
ROW3
END
```

Session 20
Task Cards for the Logo Problem Box, Page 4

Study this design. Can you find the pattern? Write the procedures that Turtle will need to draw the design on the screen.



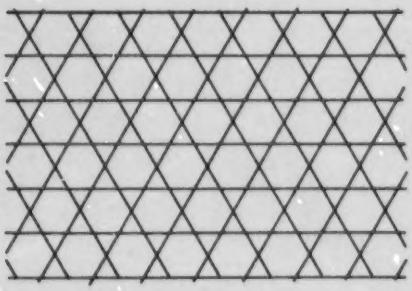
Study this design. Find the pattern that repeats. One pattern must not overlap the pattern on either side. The second row of this pattern must interlock with the row above it so that there are no "empty spaces" on the screen.



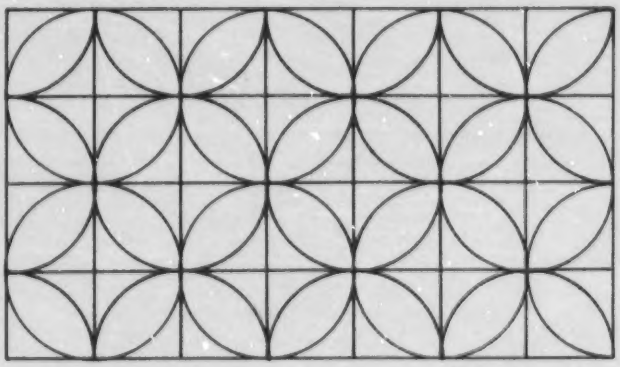
© ICCE 1984. Reproduction for classroom use permitted.

Session 20
Task Cards for the Logo Problem Box, Page 5

Study this design. Can you find the pattern? Write the procedures to draw the design on the screen.



Study this design. Find the smallest part that can be used as the pattern. This smallest part is then rotated to make the larger pattern.

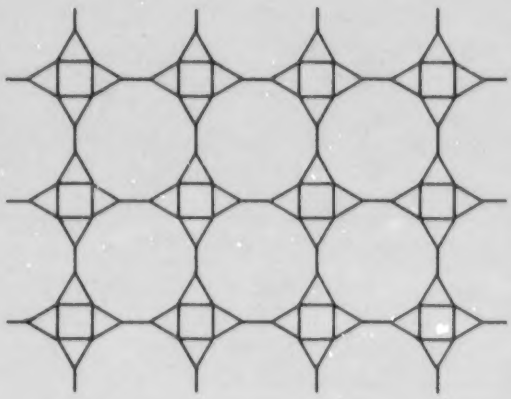


© ICCE 1984. Reproduction for classroom use permitted.

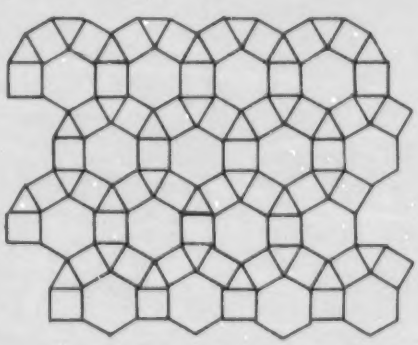
Session 20

Task Cards for the Logo Problem Box, Page 6

Study this design. Can you find the pattern? Can you write procedures so that Turtle will draw this design on the screen in a way that will allow all the patterns to interlock perfectly without leaving any gaps? The patterns must not overlap.



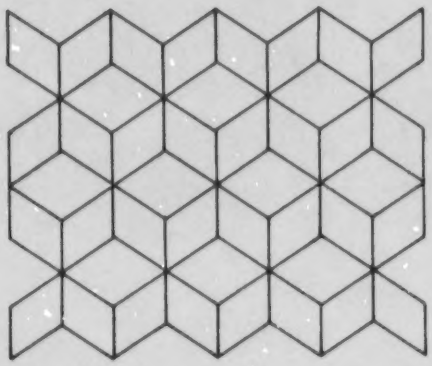
Study this pattern carefully before writing the procedures. Remember: There can be no areas not enclosed by a polygon, and polygons cannot overlap. This design is more difficult because each row seems to mesh with the preceding row.



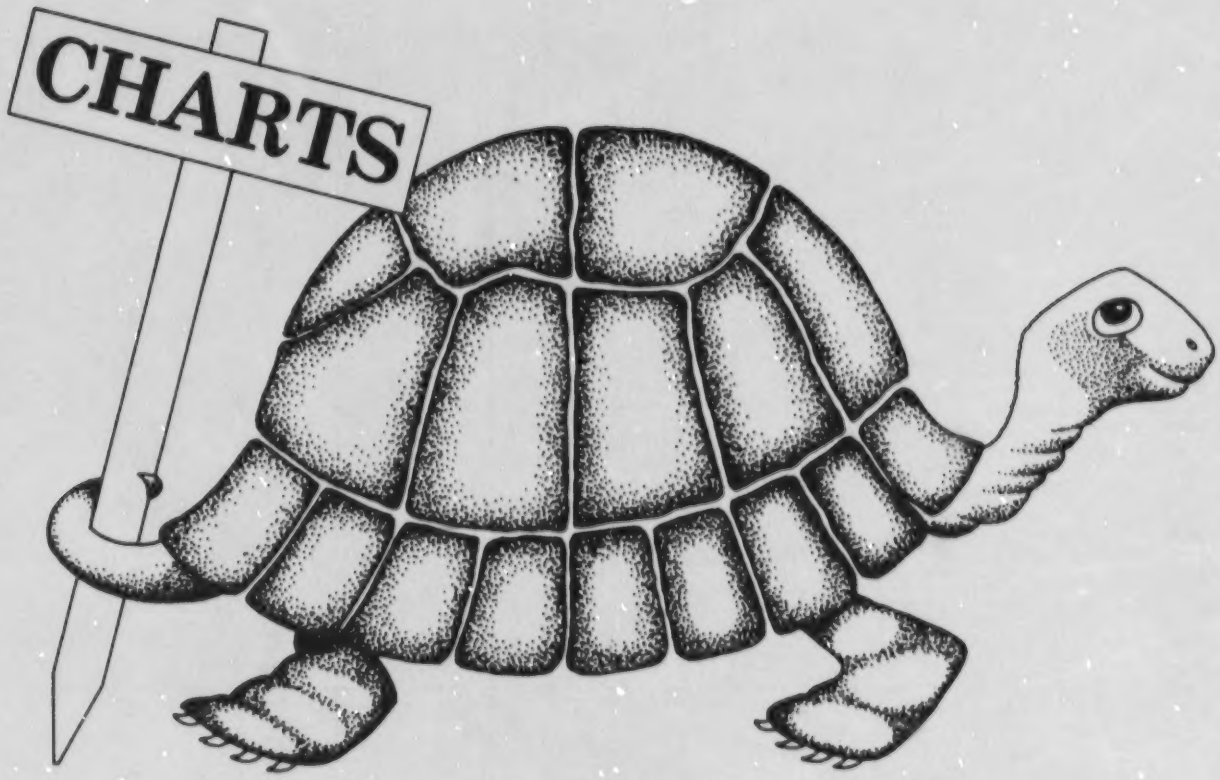
© ICCE 1984. Reproduction for classroom use permitted.

Session 20
Task Cards for the Logo Problem Box, Page 7

Study this design. How many different ways can you fill the screen with this design? Each way must interlock with no gaps!



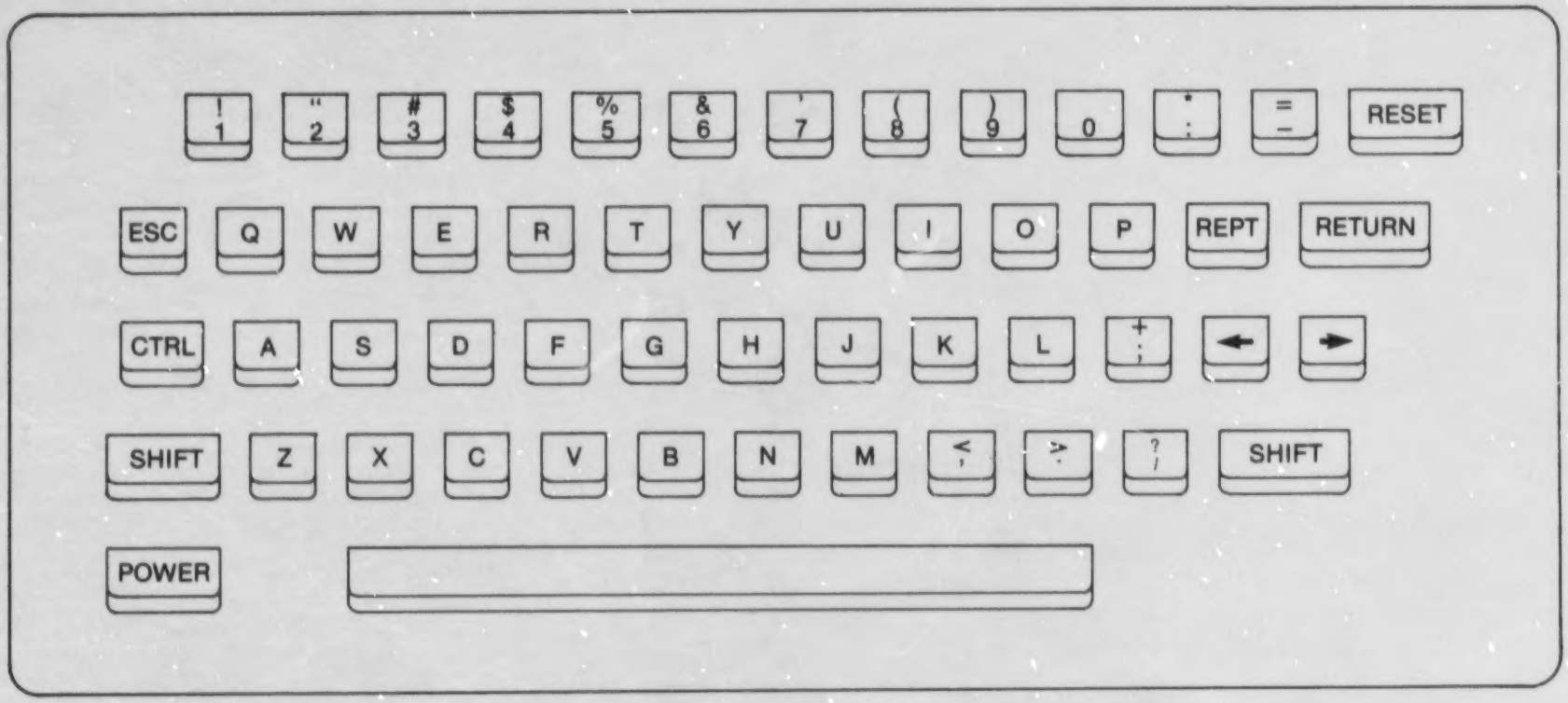
© ICCE 1984. Reproduction for classroom use permitted.


CHARTS

© ICCE 1984. Reproduction for classroom use permitted.

KEYBOARD

Apple II



" ← UPPER FACE: HOLD DOWN SHIFT. THEN PRESS THE KEY.
2 ← LOWER FACE: PRESS KEY.

162

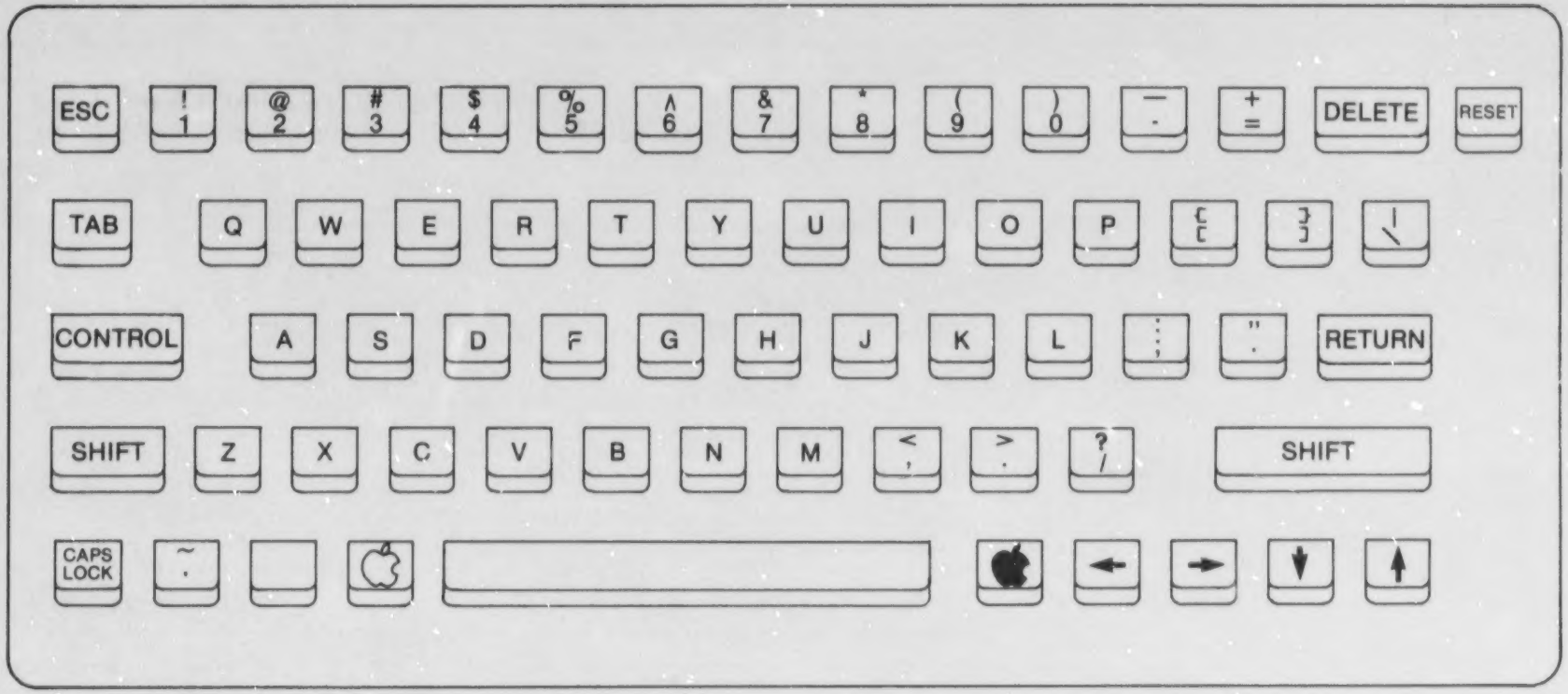
SHIFT M →]
SHIFT N → [

163

© ICCE 1984. Reproduction for classroom use permitted.

KEYBOARD

Apple IIe

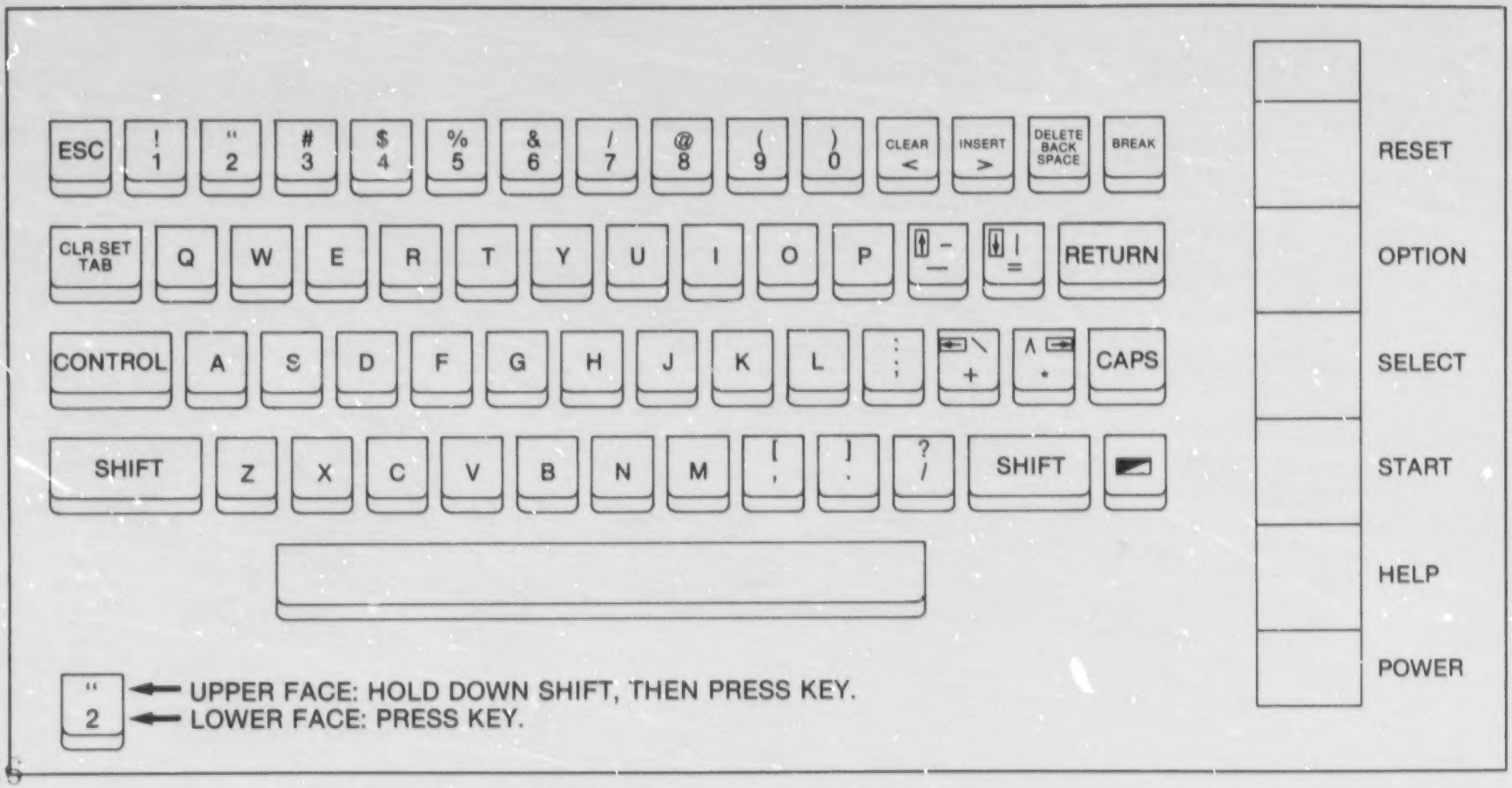


@ ← UPPER FACE: HOLD DOWN SHIFT. THEN PRESS THE KEY.
2 ← LOWER FACE: PRESS KEY.

© ICCE 1984. Reproduction for classroom use permitted.

Atari

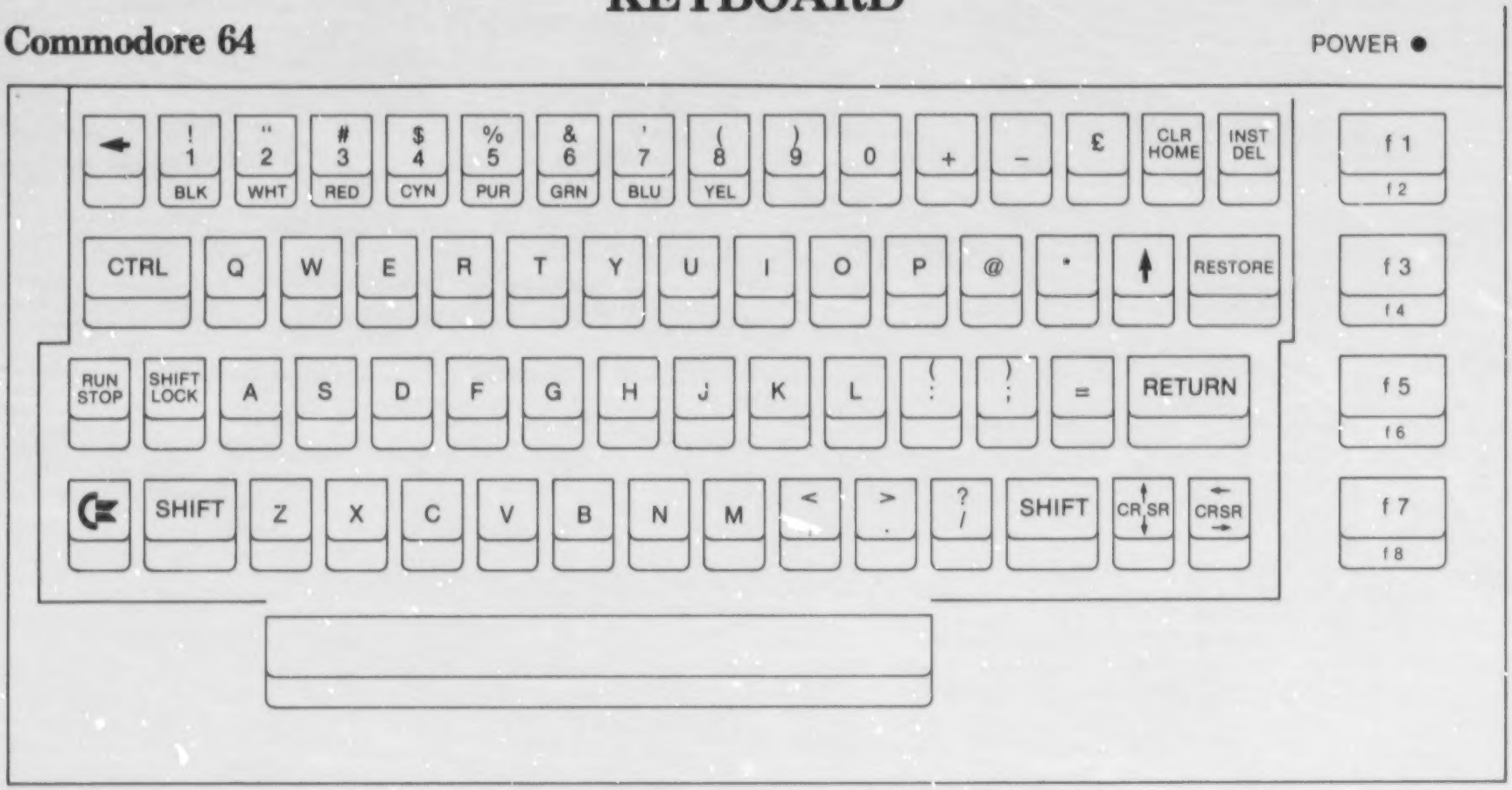
KEYBOARD



© ICCE 1984. Reproduction for classroom use permitted.

KEYBOARD

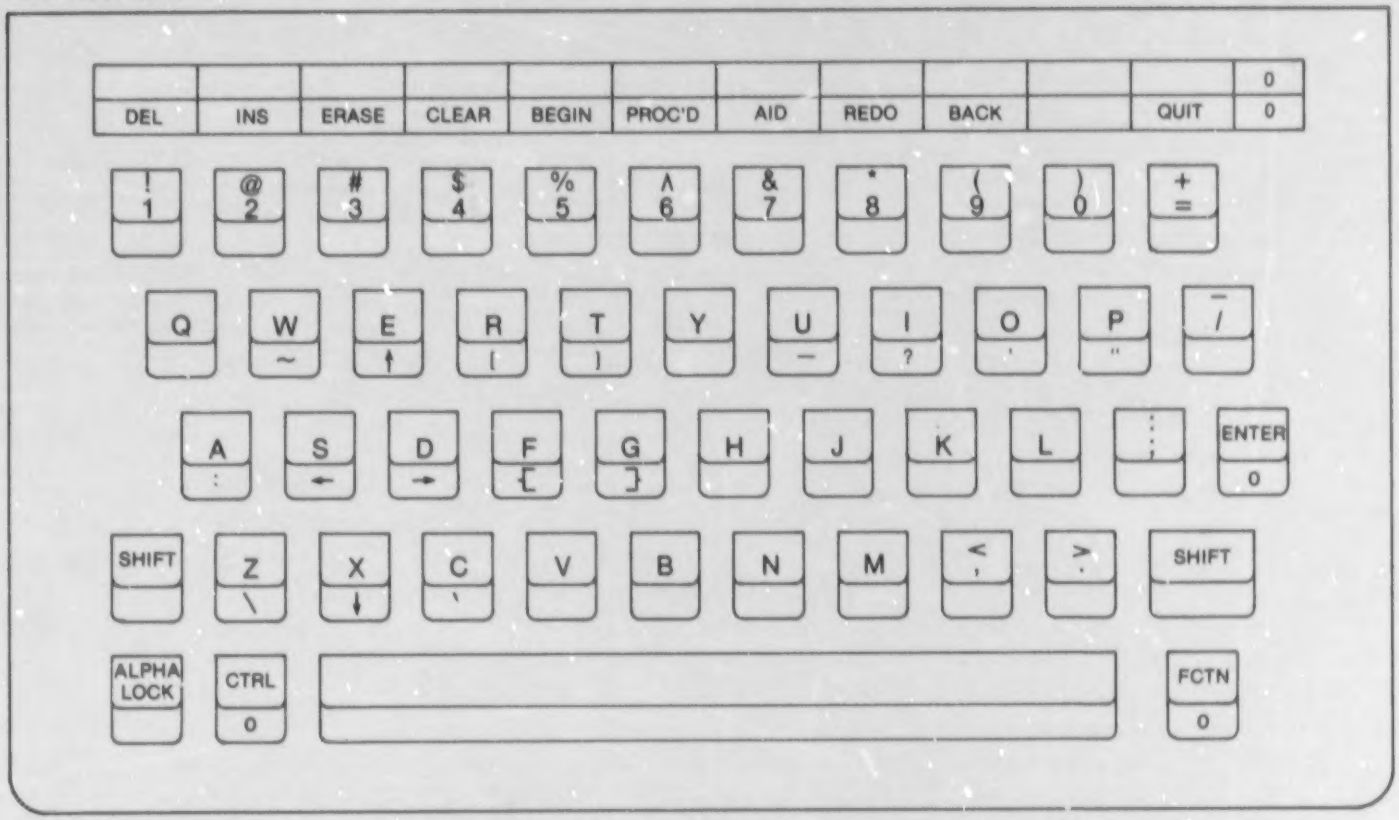
Commodore 64



" ← UPPER FACE: HOLD DOWN SHIFT. THEN PRESS KEY.
2 ← LOWER CASE: PRESS KEY.

© ICCE 1984. Reproduction for classroom use permitted.

TI 99/4A KEYBOARD


DEL INS ERASE CLEAR BEGIN PROC'D AID REDO BACK QUIT 0
1 2 3 4 5 6 7 8 9 0 =
Q W E R T Y U I O P /
A S D F G H J K L ; ENTER
SHIFT Z X C V B N M , . SHIFT
ALPHA LOCK CTRL FCTN

@ 2 ← UPPER FACE: HOLD DOWN SHIFT. THEN PRESS THE KEY.
← LOWER FACE: PRESS KEY.

171

E ↑ ← FRONT: HOLD DOWN FCTN. THEN PRESS THE KEY YOU WANT TYPED.

170

DEL INS ERASE

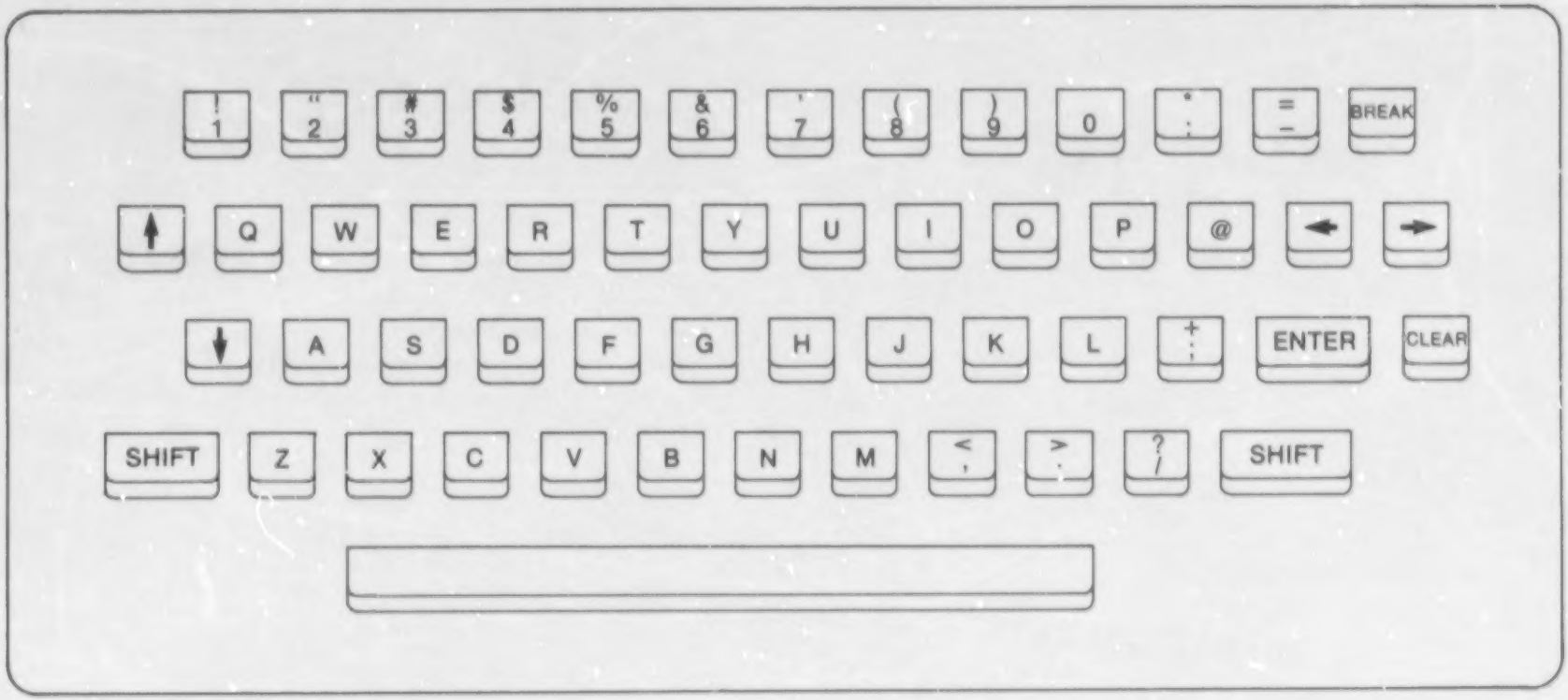
FCTN ← HOLD DOWN FCTN KEY.

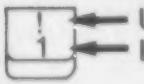
3 ← THEN PRESS KEY BELOW THE DESIRED ACTION
(DEL, INS, ERASE, CLE[illegible] etc.) For example, to erase, hold down FCTN and

© ICCE 1984. Reproduction for classroom use permitted.

KEYBOARD

TRS-80 Color Computer





← UPPER FACE: HOLD DOWN SHIFT. THEN PRESS KEY.
← LOWER FACE: PRESS KEY.

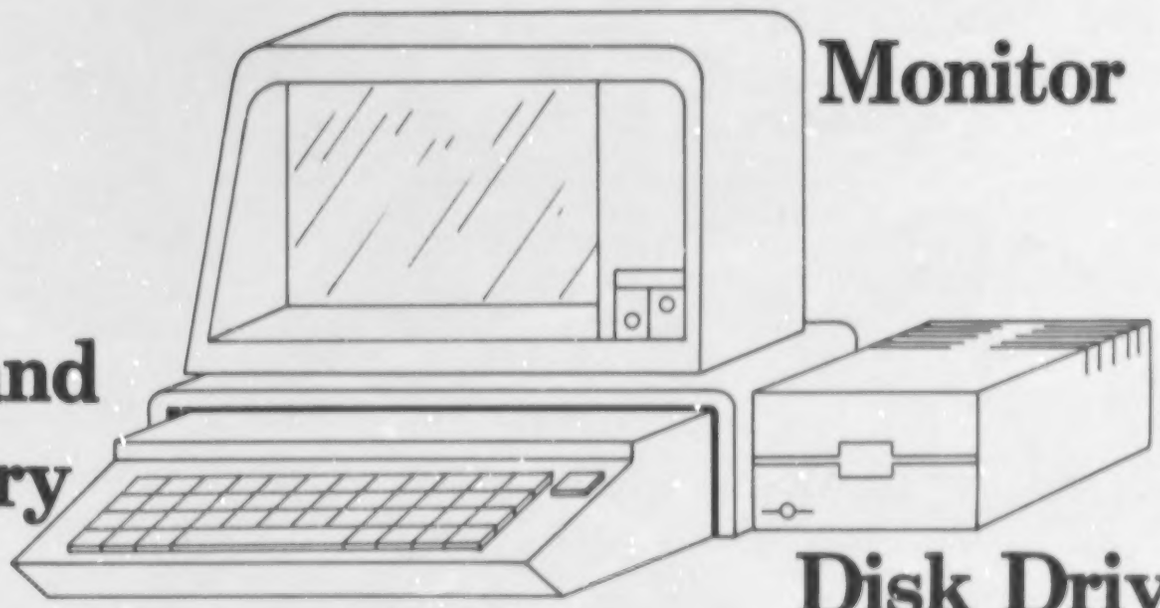
172

173

© ICCE 1984. Reproduction for classroom use permitted.

COMPUTER SYSTEM

Apple II
Apple IIe



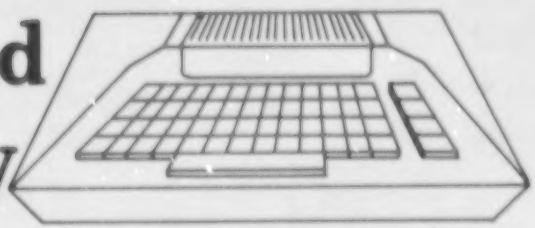
© ICCE 1984. Reproduction for classroom use permitted.

COMPUTER SYSTEM

Atari

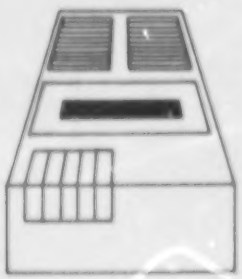
Monitor

CPU and Memory



Keyboard

Disk Drive



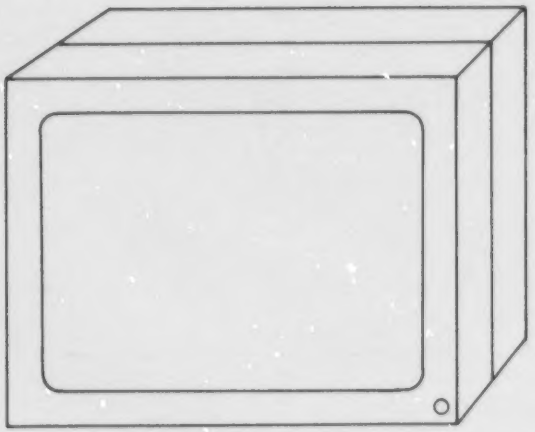
176

177

© ICCE 1984. Reproduction for classroom use permitted.

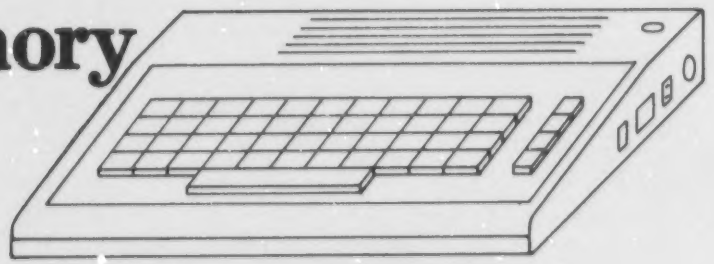
COMPUTER SYSTEM

Commodore 64

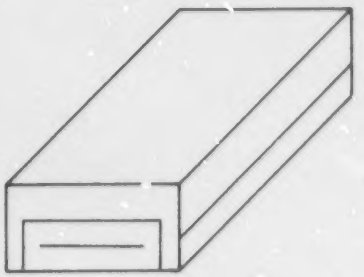


Monitor

CPU and Memory



Keyboard

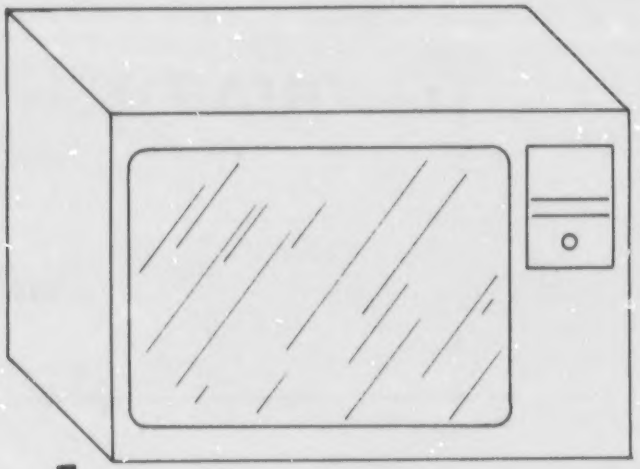


Disk Drive

© ICCE 1984. Reproduction for classroom use permitted.

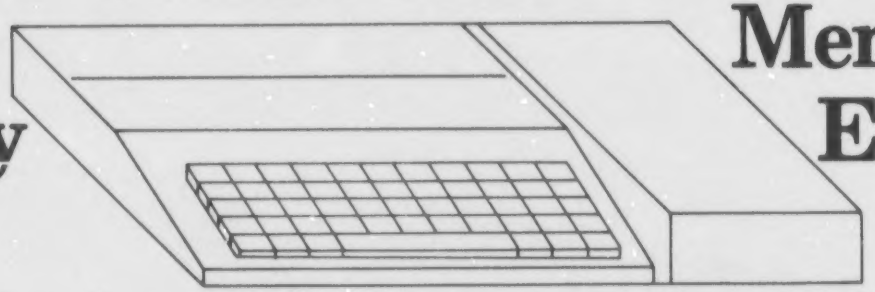
TI 99/4A

COMPUTER SYSTEM



Monitor

CPU and Memory



Memory Expansion

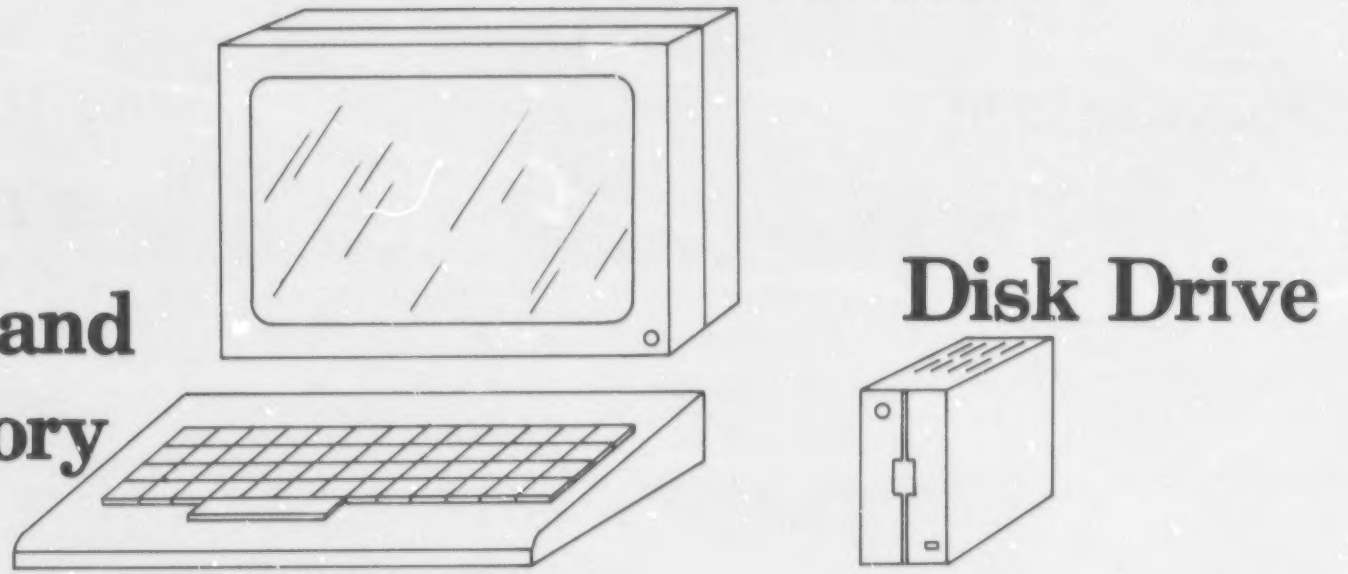
Keyboard

Logo Module

© ICCE 1984. Reproduction for classroom use permitted.

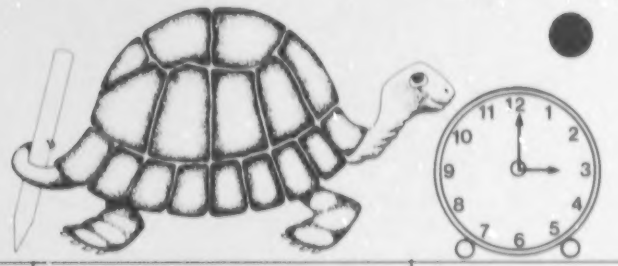
COMPUTER SYSTEM

TRS-80 Color Computer



© ICCE 1984. Reproduction for classroom use permitted.

SIGN UP FOR TIME-ON



MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY

TIME-ON Chart

Suggested Use:

Enlarge to approximately 71 x 56 cm (22"x28"), using an opaque projector or by making a thermal transparency and projecting with an overhead projector.

Laminate the chart.

Use a non-permanent marker to fill in the blocks according to the available time slots for each day:

8-8:30
8:30-9
9-9:30 —LIBRARY—
9:30-10

Block out any time slots unavailable to students in the daily schedule (library time, P.E. time, lunch, etc.)

Let students sign up for 1/2 hour time slots. A system should be set up to rotate by rows or alphabetically by name so that the same students do not get the first choice every week. Students should only select one 1/2 hour slot until all students have had a turn. If there are sufficient slots left, a second choice can then be made. Students may nave to share the second time slot.

THURSDAY
8-8:30 Sarah
8:30-9 Mike M.
9-9:30 —LIBRARY—
9:30-10 Eric
10-10:30 Kris H.
10:30-11 Peter

© ICCE 1984. Reproduction for classroom use permitted.

PARTNER Chart

To Pals • To Buddies • To Friends • To Partners
[Collect Assignments • Help Debug • Share Ideas •
Save Papers • Help Solve Problems]

© ICCE 1984. Reproduction for classroom use permitted.

PARTNER Chart

Suggested Use:

Use a sociogram to set up a system for selecting partners, or select pairs according to your knowledge of which students work well together. Ask students to list three classmates whom they would like as a partner. Explain that they might not get their first or second choice. The students should understand that they will not be able to share computer time with their partners.

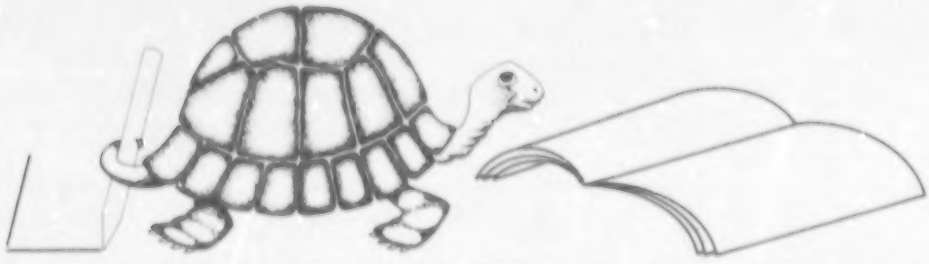
Explain that the partner's tasks are as follows:

1. Write down any missed assignments in all subject areas while the partner is away from regular class at the computer (or absent from school). BE THOROUGH. List complete information such as problem numbers, page numbers and when the work is due.
2. Collect any materials, notes, ditto sheets or other information that may be passed out while the partner is at the computer.
3. Hand in assignments that are due.
4. Do any *minor* explaining of missed assignments and school work. Note: The teacher or teacher aide should explain any new concepts.

Laminate the PARTNER chart. Use non-permanent markers to list the partner names. Both partners' names are put on one line. Post the chart in a place that students can refer to easily.

© ICCE 1984. Reproduction for classroom use permitted.

PRIMITIVE COMMANDS



© ICCE 1984. Reproduction for classroom use permitted.

PRIMITIVE COMMANDS Wall Chart

Suggested Use:

Enlarge to approximately 71x56 cm (22"x28") using an opaque projector or by making a thermal transparency and projecting with an overhead projector.

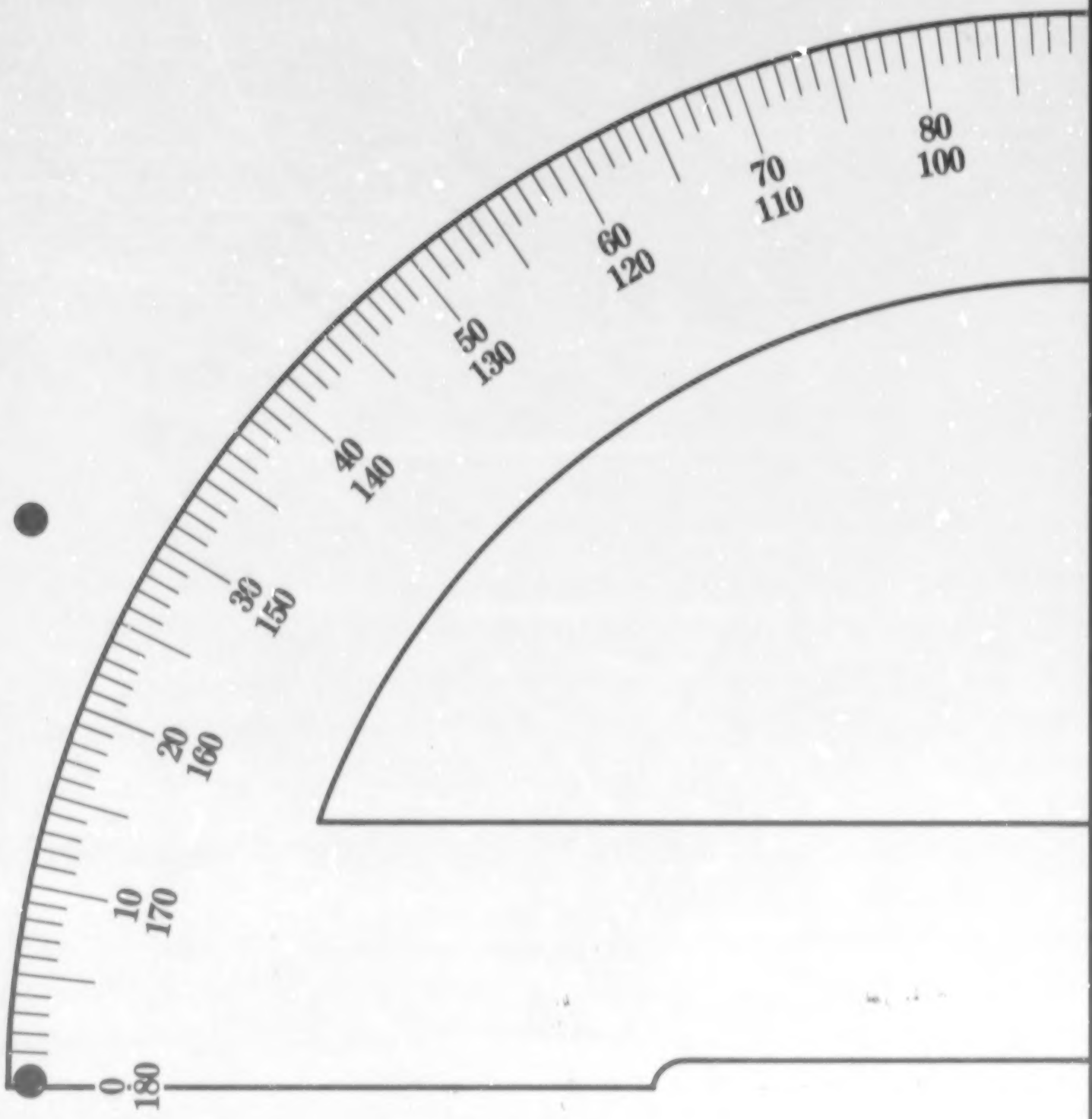
Laminate.

As the primitives are introduced, use a non-permanent marker to write each one on the chart. As earlier primitives become internalized, wash off and replace with unfamiliar and recently introduced commands.

PRIMITIVE COMMANDS		
FORWARD	FD	
BACK	BK	
RIGHT	RT	
LEFT	LT	

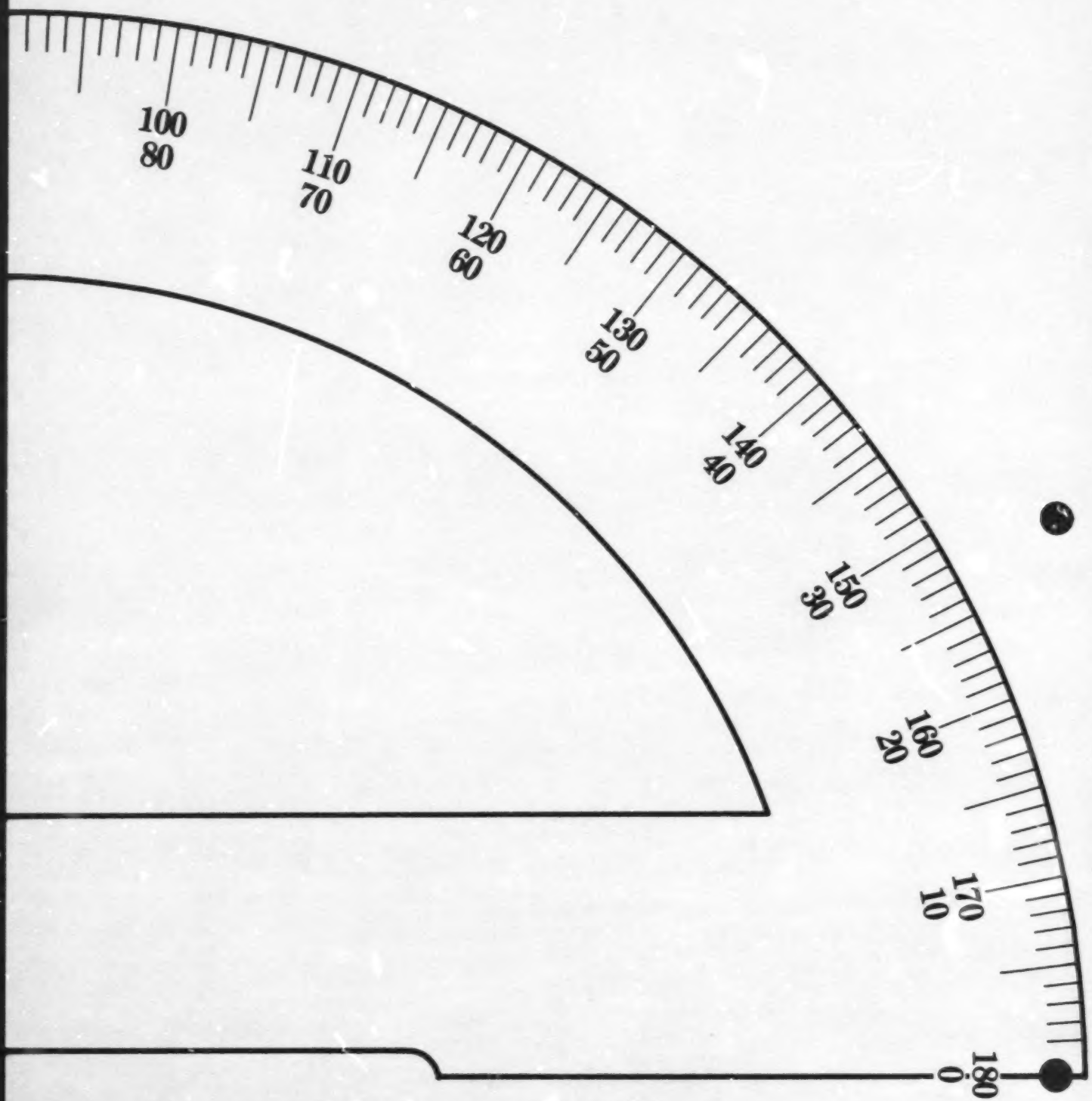
© ICCE 1984. Reproduction for classroom use permitted.

LARGE



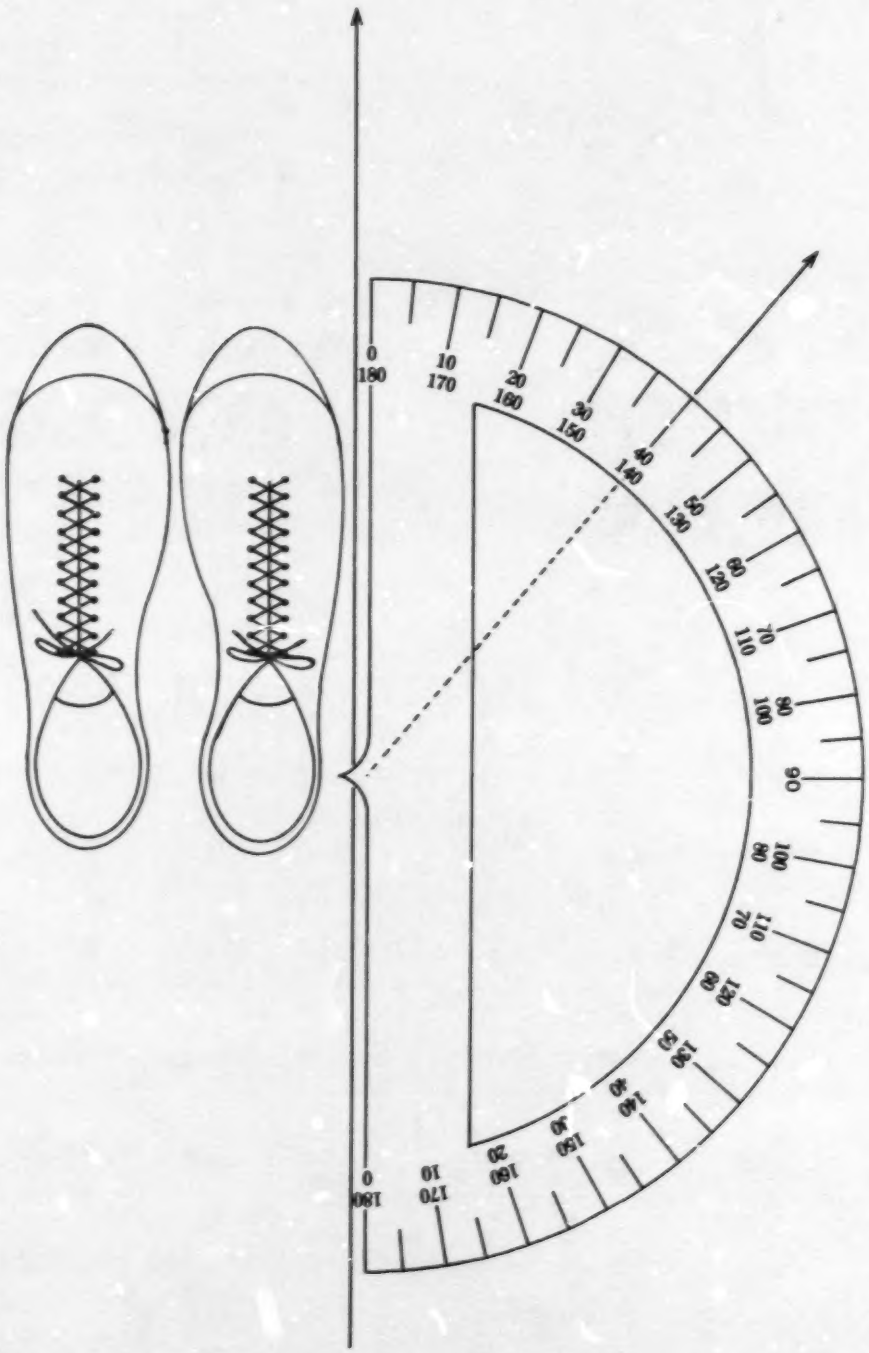
© ICCE 1984. Reproduction for classroom use permitted.

PROTRACTOR



© ICCE 1984. Reproduction for classroom use permitted.

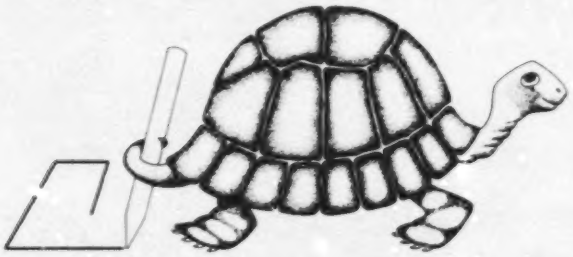
USING THE LARGE PROTRACTOR
Wall Chart

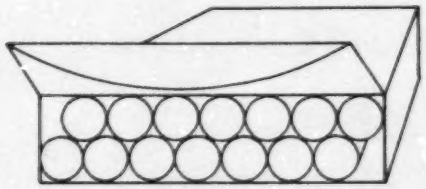


© ICCE 1984. Reproduction for classroom use permitted.

COLOR CHART

Apple Logo(SETBG, SETPC)
MIT Logo(BG, PC)





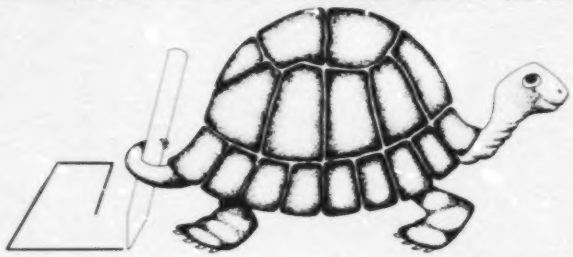
0 BLACK
1 WHITE
2 GREEN
3 VIOLET
4 ORANGE
5 BLUE

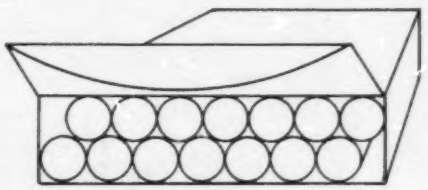
Pen Color	ON THIS BACKGROUND:						
	0	1	2	3	4	5	6
0 shows →	Black	Black	Black	Black	Black	Black	Black
1 →	White	White	White	White	White	White	White
2 →	Green	Green	Green	Green	Orange	Orange	White
3 →	Violet	Violet	Violet	Violet	Blue	Blue	White
4 →	Orange	Orange	Green	Green	Orange	Orange	White
5 →	Blue	Blue	Violet	Violet	Blue	Blue	White

© ICCE 1984. Reproduction for classroom use permitted.

COLOR CHART

Atari Logo (SETBG, SET PC)





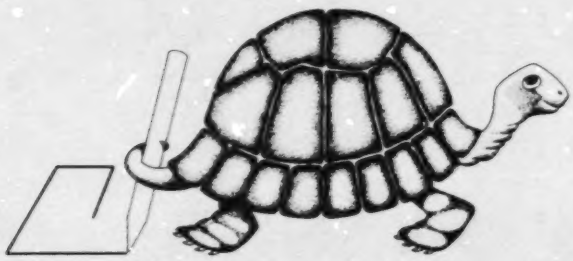
0-7	GRAY
8-15	LIGHT ORANGE (GOLD)
16-23	ORANGE
24-31	RED-ORANGE
32-39	PINK
40-47	PURPLE
48-55	PURPLE-BLUE
56-63	BLUE
64-71	BLUE
72-79	LIGHT BLUE
80-87	TURQUOISE
88-95	GREEN-BLUE
96-103	GREEN
104-111	YELLOW-GREEN
112-119	ORANGE-GREEN
120-127	LIGHT ORANGE

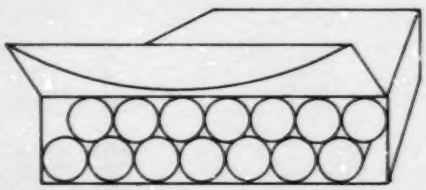
Note: For each color, the lowest number is the darkest shade and the highest number the lightest shade of the color. For example: 0 is black and 7 is white.

© ICCE 1984. Reproduction for classroom use permitted.

COLOR CHART

COMMODORE 64 (BG, PC)



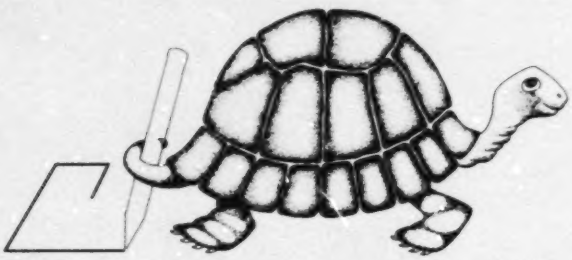


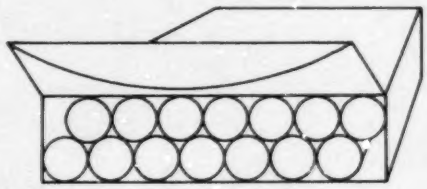
PENCOLOR (PC)		BACKGROUND (BG)	
NUMBER	SHADE OF	NUMBER	SHADE OF
0	BLACK	8	ORANGE
1	WHITE	9	BROWN
2	RED	10	LT RED
3	CYAN	11	DK GRAY
4	PURPLE	12	MED GRAY
5	GREEN	13	LT GREEN
6	BLUE	14	LT BLUE
7	YELLOW	15	LT GRAY

© ICCE 1984. Reproduction for classroom use permitted.

COLOR CHART

TI (CB, SC)





SETCOLOR SC

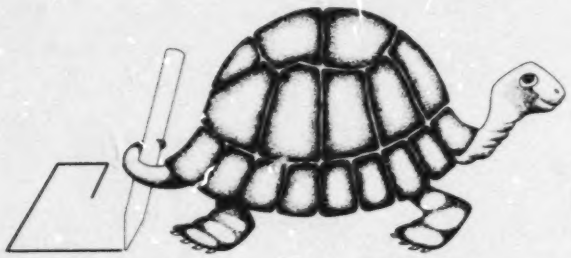
COLORBACKGROUND (CB)

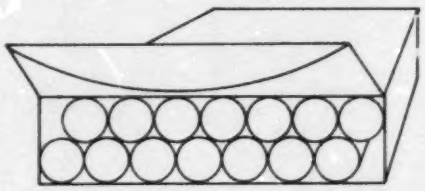
NUMBER	COLOR	NUMBER	COLOR
0	CLEAR	8	RUST
1	BLACK	9	ORANGE
2	GREEN	10	YELLOW
3	LIME	11	LEMON
4	BLUE	12	OLIVE
5	SKY	13	PURPLE
6	RED	14	GRAY
7	CYAN	15	WHITE

© ICCE 1984. Reproduction for classroom use permitted.

COLOR CHART

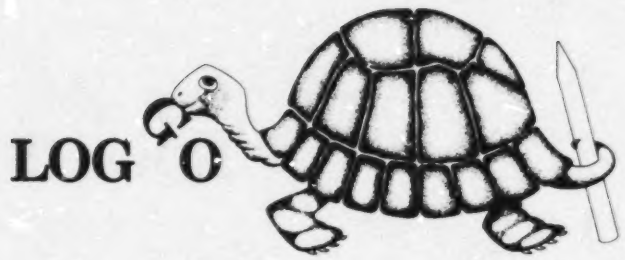
TRS-80 Color Computer (BG, PC)





0 BLACK
1 GREEN
2 YELLOW
3 BLUE
4 RED
5 BUFF
6 CYAN
7 MAGENTA
8 ORANGE

© ICCE 1984. Reproduction for classroom use permitted.



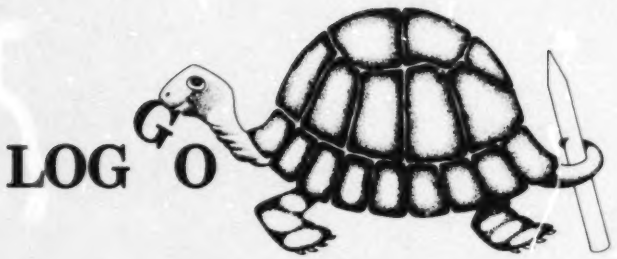
Apple Logo
Apple II+
Apple IIe

Key Functions for the Edit Mode

Hold Down	Then Press	Function
	Left Arrow ←	Erasees the character to the left of the cursor and moves the cursor one space to the left.
CTRL	A	Moves cursor to the beginning of the line.
CTRL	B	Moves cursor back one space.
CTRL	F	Moves cursor forward one space.
	Right Arrow →	Moves cursor forward one space (right one space).
CTRL	C	Leaves the edit mode; processes the new procedure and returns to command mode.
CTRL	D	Deletes the character under the cursor.
CTRL	E	Moves the cursor to the end of the line.
CTRL	G	Exits the edit mode without processing the procedure.
CTRL	K	Deletes (kills) all the characters from the cursor to the end of the line.
CTRL	N	Moves the cursor down one line.
CTRL	O	Opens a new line at the cursor position.
CTRL	P	Moves the cursor up one line.

© ICCE 1984. Reproduction for classroom use permitted.

Atari Logo

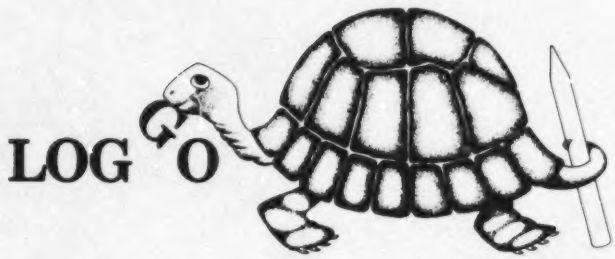


Key Functions for the Edit Mode

To edit the procedure SQUARE, type: EDIT "SQUARE

Hold Down	Then Press	Function
CTRL	↑	Moves cursor up one line.
CTRL	↓	Moves cursor down one line.
CTRL	←	Moves cursor left one line.
CTRL	→	Moves cursor right one line.
CTRL	A	Moves cursor to the **beginning** of the line.
CTRL	E	Moves cursor to the **end** of the line.
CTRL	X	Moves cursor to the **beginning** of the editor.
CTRL	Z	Moves cursor to the **end** of the editor.
	DELETE BACK S	Erases character to the left of cursor.
CTRL	DELETE BACK S	Erases the character at the cursor.
SHIFT	DELETE BACK S	Deletes from the cursor position to the end of the line. The text goes into a buffer that holds 110 characters.
CTRL	Y	Returns the text that is in the buffer.
	RETURN	Moves the cursor to beginning of a new line.
CTRL	INSERT	Makes a new line at the position of the cursor but doesn't move cursor.
CTRL	V	Scrolls to the next page in editor.
CTRL	W	Scrolls back to the previous page in editor.
CTRL	1	Stops scrolling until CTRL 1 pressed again.
	ESC	Usual way to exit from the editor.
	BREAK	Exits the editor with procedure unchanged.

© ICCE 1984. Reproduction for classroom use permitted.

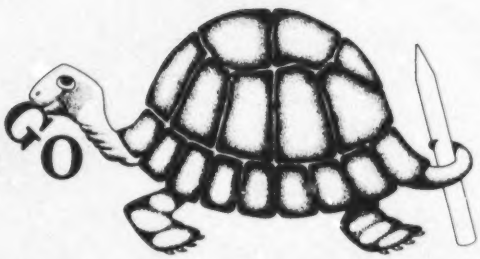


MIT Logo
Apple II+
Apple IIe

Key Functions for the Edit Mode

Hold Down	Then Press	Function
	ESC (DEL)	Erases the character to the left of the cursor and moves the cursor one space to the left.
	Arrow Keys	Moves the cursor one character in the direction of the arrow without erasing the character. Apple II: The REPT key must be held down with the arrow key. The REPT key can be held down with the arrow key for continuous erasing. Apple IIe: The arrow keys will repeat if held down. DO NOT use the . The will delete the line.
REPT	Any Key	Makes any character keep repeating (Apple II only).
CTRL	A	Moves the cursor to the beginning of the line.
CTRL	B	Moves the cursor one screen full of text backward.
CTRL	C	Leaves the edit mode; processes the new procedure and returns to command mode.
CTRL	D	Deletes the character under the cursor.
CTRL	E	Moves the cursor to the end of the line.
CTRL	F	Moves the cursor one screen full of text forward.
CTRL	G	Exits the edit mode without processing the procedure.
CTRL	K	Deletes (kills) all the characters from the cursor to the end of the line.
CTRL	N	Moves the cursor down one line.
CTRL	O	Opens a new line at the cursor position.
CTRL	P	Moves the cursor up one line.

© ICCE 1984. Reproduction for classroom use permitted.

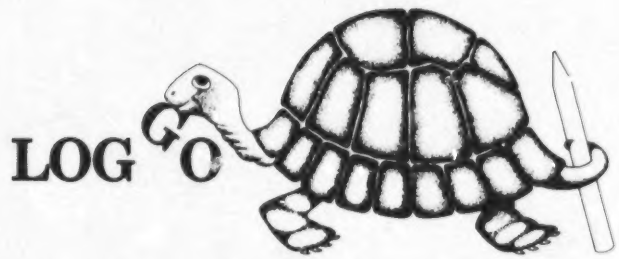


MIT Logo
Commodore 64

Key Functions for the Edit Mode

Hold Down	Then Press	Function
	ESC (DEL)	Erases the character to the left of the cursor and moves the cursor one space to the left.
	Arrow Keys	Moves the cursor one character in the direction of the arrow without erasing the character. The arrow keys will repeat if held down.
CTRL	A	Moves the cursor to the beginning of the line.
CTRL	B	Moves the cursor one screen full of text backward.
CTRL	C	Leaves the edit mode; processes the new procedure and returns to command mode.
CTRL	D	Deletes the character under the cursor.
CTRL	L	Moves the cursor to the end of the line.
CTRL	F	Moves the cursor one screen full of text forward.
CTRL	G	Exits the edit mode without processing the procedure.
CTRL	K	Deletes (kills) all the characters from the cursor to the end of the line.
CTRL	N	Moves the cursor down one line.
CTRL	O	Opens a new line at the cursor position.
CTRL	P	Moves the cursor up one line.

© ICCE 1984. Reproduction for classroom use permitted.

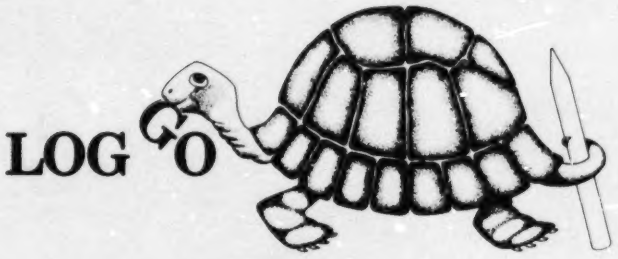


TI Logo

Key Functions for the Edit Mode

Hold Down	Then Press	Function
FCTN	5 (BEGIN)	Moves the cursor to the beginning of the line.
FCTN	6 (PROC'D)	Moves the cursor to the end of the line.
FCTN	E ↑	Moves the cursor up one line.
FCTN	S ←	Moves the cursor one space to the left.
FCTN	D →	Moves the cursor one space to the right.
FCTN	X ↓	Moves the cursor down one line.
FCTN	ENTER	Creates a blank row between the line the cursor is on and the next line if the cursor is at the end of a line.
FCTN	ERASE	Moves the entire line up one line if the cursor is under the first character of a line.
FCTN	ERASE	Erases the character or space one space to the left of the cursor.
FCTN	1 (DEL)	Erases the character or space above the cursor. Moves the next line up to that line if the cursor is at the end of the line.
FCTN	4 (CLEAR)	Erases the character or space above the cursor and everything to the right of the cursor.
FCTN	9 (BACK)	Leaves the edit mode; processes the new procedure and returns to the immediate mode.

© ICCE 1984. Reproduction for classroom use permitted.



Key Functions for the Edit Mode

Hold Down	Then Press	Function
	BREAK Key then E Key	Enters edit mode.
	ENTER Key	Moves the text up one line on the screen; if it is already on the last line, a new line is added to the text end.
	↑ Up Arrow	Moves the text up one line.
	↓ Down Arrow	Moves the text down one line.
	Left Arrow ←	Moves the cursor left one character unless it is already at the beginning of the line.
	Right Arrow →	Moves the cursor right one character unless it is at the line end.
	CLEAR Key	Moves to the top line of the text.
SHIFT	↑ Up Arrow	Scrolls the text up continuously until any key is pressed.
SHIFT	↓ Down Arrow	Inserts a blank line in front of the current line if the cursor is in column 1 (the current line bumps down off the screen); if not in column 1, then the current line is split at the cursor location into two lines.
SHIFT	Left Arrow ←	Deletes the character under the cursor and moves the remainder of the line left to close the gap. If the line is blank, it is removed.
SHIFT	Right Arrow →	Inserts a blank into the line at the cursor location by moving the remainder of the line right one space.
	BREAK Key	Exits edit mode and returns to break mode.

© ICCE 1984. Reproduction for classroom use permitted.

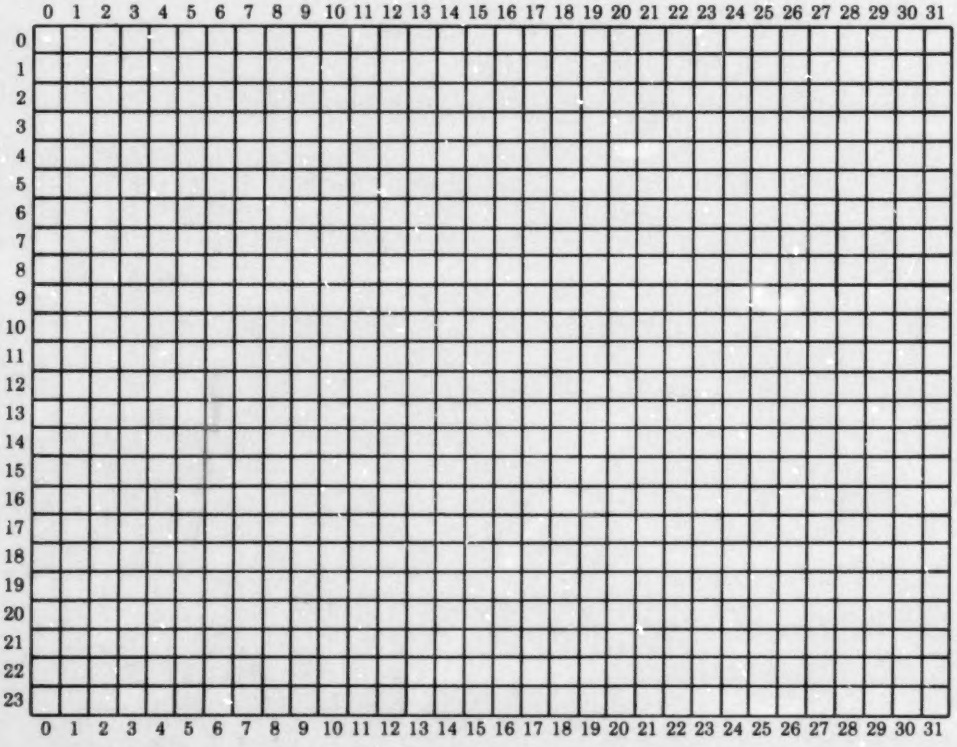
PUTTILE (PT) COMMAND

PUTTILE 65 14 9

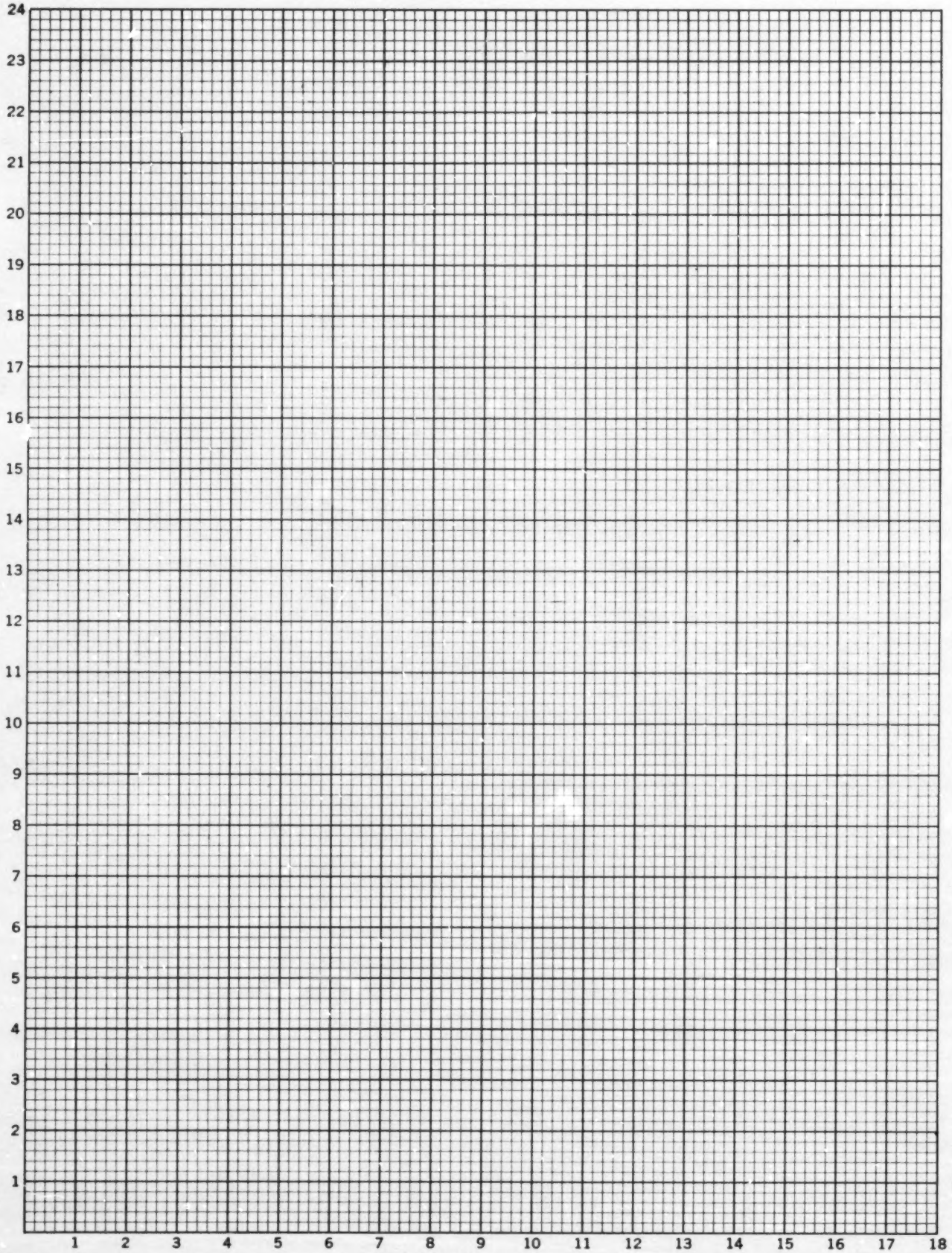
Character	Column	Row

32	(Space)	48	0	64	@	80	P
33	!	49	1	65	A	81	Q
34	"	50	2	66	B	82	R
35	#	51	3	67	C	83	S
36	$	52	4	68	D	84	T
37	%	53	5	69	E	85	U
38	&	54	6	70	F	86	V
39	'	55	7	71	G	87	W
40	(	56	8	72	H	88	X
41	)	57	9	73	I	89	Y
42	*	58	:	74	J	90	Z
43	+	59	;	75	K	91	[
44	,	60	<	76	L	92	\
45	-	61	=	77	M	93	]
46	.	62	>	78	N	94	^
47	/	63	?	79	O	95	_

EACH SQUARE IS 8 TURTLE STEPS ON THE SCREEN



© ICCE 1984. Reproduction for classroom use permitted.



© ICCE 1984. Reproduction for classroom use permitted.

References

Books

Abelson, Harold. *Logo for the Apple II.* Peterborough, NH: Byte-McGraw Hill, 1982.

Abelson, Harold, and Andrea diSessa. *Turtle Geometry: The Computer as a Medium for Exploring Mathematics.* Cambridge, MA: The MIT Press, 1981.

Copeland, Richard W. *How Children Learn Mathematics: Teaching Implications of Piaget's Research.* New York: Macmillan Publishing Co., Inc., 1974.

Papert, Seymour, Daniel Watt, Andrea diSessa, Sylvia Weir. *Final Report of Brookline LOGO Project, Parts II and III.* Artificial Intelligence Lab, Massachusetts Institute of Technology, Cambridge, MA, 1979.

Papert, Seymour. *Mindstorms: Children, Computers and Powerful Ideas.* New York: Basic Books, Inc., 1980.

Phillips, John L., Jr. *The Origins of Intellect, Piaget's Theory.* W. H. Freeman & Co., San Francisco, 1975.

Piaget, Jean, Barbel Inhelder and Alina Szeminska. *The Child's Conception of Geometry.* Trans. E. A. Lunzer. New York: W. W. Norton & Co., 1981.

Piaget, Jean, and Barbel Inhelder. *The Child's Conception of Space.* New York: W. W. Norton & Co., 1966.

Raphael, Bertram. *The Thinking Computer, Mind Inside Matter.* W. H. Freeman and Company, San Francisco, CA, 1976.

Schwebel, Milton and Jane Raph. *Piaget in the Classroom.* Basic Books, Inc., New York, 1973.

Taylor, Robert P., ed. *The Computer in the School: Tutor, Tool, Tutee.* New York: Teachers College Press, Columbia University, 1980.

Magazines and Journals

Byte Magazine, Byte Subscriber Service, P.O. Box 328, Hancock, NH 03449. (Vol. 7, #8 [August 1982] issue devoted entirely to Logo.)

Compute! Magazine, P.O. Box 5406, Greensboro, NC 27403. (Apple, Atari, Commodore)

The Computing Teacher, The Journal of The International Council for Computers in Education (ICCE), University of Oregon, 1787 Agate St., Eugene, OR 97403. (Nov. 1982 [Vol. 10, #3] and Dec./Jan. 1983-84 [Vol. 11, #5] are special issues on Logo.)

80 Microcomputing Magazine, P.O. Box 981, Farmingdale, NY 11737. (TRS 80 users)

Home Computer Magazine, P.O. Box 5537, Eugene, OR 97405. (Includes a special section on Logo each month.)

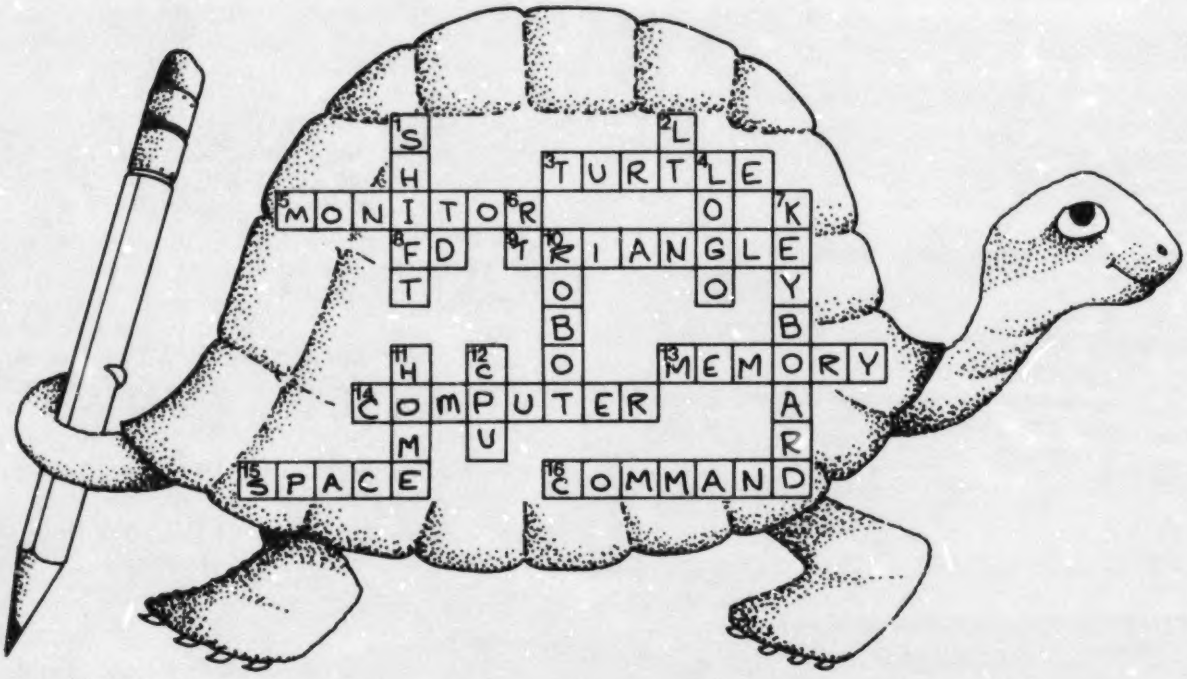
Miscellaneous

Bibliography of Logo Memos Available for Distribution. Memos from the original Logo project at Massachusetts Institute of Technology. Order the Bibliography from: MIT Publications, Artificial Intelligence Laboratory, 545 Technology Square, Room 908, Cambridge, MA 02139.

Answer Appendix

(Answers to Some Problem, Practice and Challenger Sheets)

Session 3 Problem Sheet, page



Session 4 Problem Sheet, pages

Answers will vary depending on the Logo version.

APPLE, ATARI, MIT, TRS

1. 2.

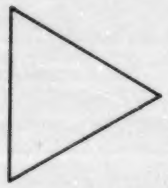
HOME moves Turtle to the center of the screen facing up. Turtle leaves a trail as it returns to HOME.

TI

1. 2.

HOME moves Turtle to the center of the screen facing up. Turtle does **not** leave a trail as it returns to HOME.

3. CLEARSCREEN (CS) or CLEAN or CLEAR
4. 360°
5. 360° 720°



Session 7 Problem Sheet, page

R=60° T=90° N=20° L=10°
E=130° U=40° P=50°

T	E	L	L	T	U	R	T	L	E
90	130	10	10	90	40	60	90	10	130

P	E	N	U	P
50	130	20	40	50

Session 9 Problem Sheet, page

There are many ways to draw each shape. The following are examples of one way for each shape.

```
TO FACE
PU HOME PD
FD 80 RT 90
FD 80 RT 90
FD 80 RT 90
FD 80 RT 90
END
```

```
TO MOUTH
PU HOME
FD 15 RT 90
FD 15 PD
FD 50 LT 90
FD 10 LT 90
FD 50 LT 90
FD 10 LT 90
PU BK 15
LT 90 BK 15
END
```

```
TO EYES
PU HOME
FD 45 RT 90
FD 15 PD
FD 15 LT 90
FD 20 LT 90
FD 15 LT 90
FD 20 LT 90
PU FD 35 PD
FD 15 LT 90
FD 20 LT 90
FD 15 LT 90
FD 20 LT 90
PU
END
```

```
TO U
PU HOME PD
FD 30 BK 30
RT 90 FD 10
LT 90 FD 30
BK 30
END
```

```
TO S
PU HOME
RT 90
FD 15 PD
FD 10 LT 90
FD 15 LT 90
FD 10 RT 90
FD 15 RT 90
FD 10
END
```

```
TO A
PU HOME
RT 90 FD 30
PD LT 90
FD 30 RT 90
FD 10 RT 90
FD 30 BK 15
RT 90 FD 10
PU FD 80
END
```

```
TO HOUSE
PU HOME PD
LT 90 FD 80
RT 90 FD 60
RT 90 BK 20
FD 200
BK 20 RT 90
FD 60 RT 90
FD 160
END
```

```
TO DOOR
BK 20 RT 90
FD 40 RT 90
FD 20 RT 90
FD 40 BK 40
END
```

```
TO WINDOW (Apple use WINDO)
LT 90 PU
FD 30 PD
FD 60 RT 90
FD 20 RT 90
FD 60 RT 90
FD 20 PU
FD 100
END
```

```
TO L
LT 90 PU
FD 50 PD
RT 90 FD 30
BK 30 RT 90
FD 20
END
```

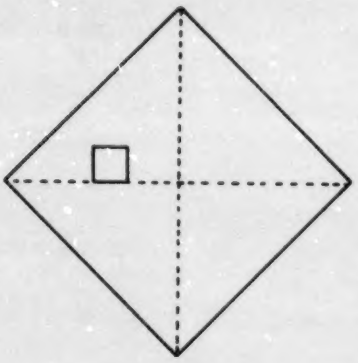
```
TO O
PU FD 5 PD
FD 20 LT 90
FD 30 LT 90
FD 20 LT 90
FD 30 LT 90
FD 20
END
```

```
TO G
PU FD 5 PD
LT 90 FD 30
RT 90 FD 30
RT 90 FD 5
PU FD 10
PD RT 90
FD 5 BK 5
LT 90 FD 15
LT 90 BK 20
FD 20
END
```

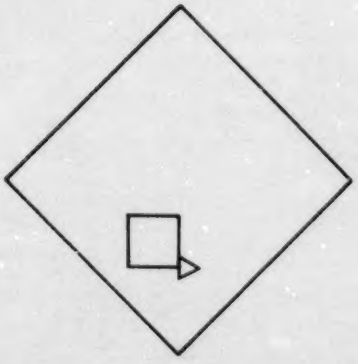
Session 10 Problem Sheet, pages

Sketches

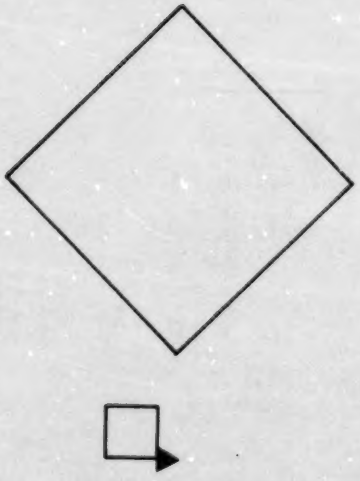
2. a.



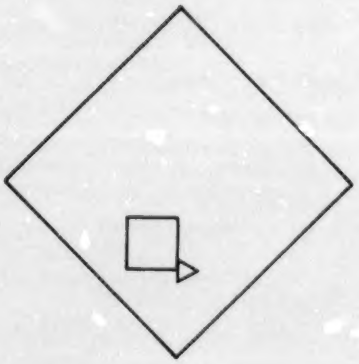
b.



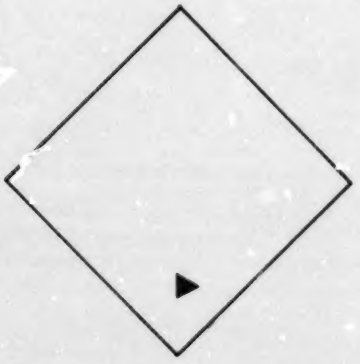
c.



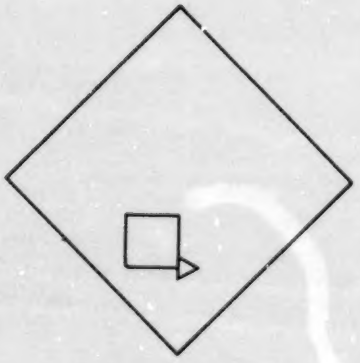
d.



e.



f.



Session 11 Problem Sheet I, page

```
TO FLAG
FD 30
RT 90
FD 30
LT 90
FD 30
RT 90→LT 90
FD 30
LT 90
FD 30
END
```

```
TO NO
FD 30
LT 45
BK 40
RT 45
FD 30
LT 90 → RT 90
FD 30
LT 90 → RT 90
FD 30
LT 90 → RT 90
FD 30
LT 90
END
```

Session 11 Problem Sheet II, page

```
TO VISIT HC'S HOUSE
FD 4
LT 30
FD 7
RT 90
FD 2
RT 90
FD 6
LT 90 → RT 90
FD 2
LT 90
FD 7
LT 90
FD 4
LT 90
FD 8
LT 90
FD 4
RT 90
FD 7
END
```

Session 12 Problem Sheet, page

The concept of subprocedures has not been specifically introduced yet. However, please note that students have been given a hint that this is possible. Again, there are many ways to make the designs given on this page. The following are only examples of one way. The following procedures use the three procedures SQUARE, HEX and OCT as listed on the Problem Sheet for Session 12.

(1)
```
TO ONE
SQUARE
FD 15 LT 90
FD 15 RT 90
SQUARE
FD 15 LT 90
FD 15 RT 90
SQUARE
FD 15 LT 90
FD 15 RT 90
SQUARE
END
```

(2)
```
TO TWO
SQUARE
FD 30 LT 90
BK 15
SQUARE
FD 30 LT 90
BK 15
SQUARE
FD 30 LT 90
BK 15
SQUARE
FD 30 LT 90
BK 15
END
```

(3)
```
TO THREE
HEX
BK 15
HEX
BK 15
HEX
END
```

(4)
```
TO FOUR
HEX
RT 90
HEX
RT 90
HEX
RT 90
HEX
RT 90
END
```

(5)
```
TO FIVE
SQUARE
RT 90 FD 15
LT 90
HEX
LT 60 BK 15
LT 60 BK 15
RT 120
SQUARE
RT 90 FD 15
LT 90
HEX
LT 60 BK 15
LT 60 BK 15
RT 120
SQUARE
END
```

(6)
```
TO SIX
SQUARE
RT 90 FD 15
LT 90
HEX
LT 60 BK 15
LT 60 BK 15
RT 120
OCT
LT 45 BK 15
LT 45 BK 15
LT 45 BK 15
RT 135
HEX
LT 60 BK 15
LT 60 BK 15
RT 120
SQUARE
END
```

Session 12 Challenger Sheet, page

1. 90°
2. 360°
3. 90°
4. 720°
5. 2
6. 90°
7. 1080°
8. 3
9. 360°
10. 5
    720°
11. If the figure closes, using only RT turns, the total turning will be a multiple of 360.

Session 13 Problem Sheet, page
APPLE, MIT

I	F		Y	O	U		L	I	K	E		T	O
1001	110	11011	11001	1111	10101	11011	1100	1001	1011	101	11011	10100	1111

C	O	D	E		T	R	Y
11	1111	100	101	11011	10100	10010	11001

F	U	L	L	S	C	R	E	E	N		A	N	D
110	10101	1100	1100	10011	11	10010	101	101	1110	11011	1	1110	100

S	P	L	I	T	S	C	R	E	E	N
10011	10000	1100	1001	10100	10011	11	10010	101	101	1110

Session 13 Problem Sheet, page
ATARI

I	F		Y	O	U		L	I	K	E		T	O
1001	110	11011	11001	1111	10101	11011	1100	1001	1011	101	11011	10100	1111

C	O	D	E		T	R	Y		F	S
11	1111	100	101	11011	10100	10010	11001	11011	110	10011

A	N	D		S	S
1	1110	100	11011	10011	10011

Session 13 Problem Sheet, page
TI

I	F		Y	O	U		L	I	K	E		T	O
1001	110	11011	11001	1111	10101	11011	1100	1001	1011	101	11011	10100	1111

C	O	D	E		T	R	Y		N	O	T	U	R	—
11	1111	100	101	11011	10100	10010	11001	11011	1110	1111	10100	10101	10010	

T	L	E		A	N	D		P	U	T	T	I	L	E
10100	1100	101	11011	1	1110	100	11011	10000	10101	10100	10100	1001	1100	101

N		C		R
1110	11011	11	11011	10010

Session 13 Problem Sheet, page
TRS

I	F		Y	O	U		L	I	K	E		T	O
1001	110	11011	11001	1111	10101	11011	1100	1001	1011	101	11011	10100	1111

C	O	D	E		T	R	Y		S	L	O	W		N
11	1111	100	101	11011	10100	10010	11001	11011	10011	1100	1111	10111	11011	1110

Session 14 Problem Sheet, page

2. 45°

3. 10

4. TO SQUARE (one possible solution)
```
REPEAT 4 [ FD 5 PU FD 5 RT 90 ] *
END
```
*TRS—Use parentheses instead of brackets.

5. APPLE, ATARI, MIT
```
TO COMB
REPEAT 6 [ RT 90 FD 30 LT 90 FD 30 BK 30 ]
END

TO SLIDES
REPEAT 4 [ FD 30 RT 135 FD 42 LT 135 ]
END
```

5. TI, TRS
```
TO COMB
REPEAT 6 [ RT 90 FD 10 LT 90 FD 10 BK 10 ]*
```
*TRS—Use parentheses instead of brackets.

6. Challenger (several possibilities):
```
TO CURVE
RT 45
REPEAT 90 [ FD 1 RT 1 ] *
END

TO CURVE
RT 45
REPEAT 45 [ FD 1 RT 2 ] *
END

TO CURVE
RT 45
REPEAT 9 [ FD 5 RT 10 ] *
END
```
*TRS—Use parentheses instead of brackets.

Session 15 Practice Sheet, page

Actual	90°	55°	130°	40°
Actual	0°	155°	20°	
Actual	215°	350°		

Session 16 Problem Sheet, page

1. TO STAIRS
```
FD 10
RT 90
FD 10
LT 90
STAIRS
END
```

2. TO TRI
```
REPEAT 3 [ FD 30 RT 120 ] *
END
```
*TRS—Use parentheses instead of brackets.

3. TO SPIN
```
TRI
LT 90
SPIN
END
```

Session 16 Challenger Sheet, page
APPLE, ATARI, MIT

```
TO H.C.DIZZY
REPEAT 4 [ FD 40 RT 90 ]
LT 45
H.C. DIZZY
END
```

Session 16 Challenger Sheet, page
TI, TRS

```
TO H.C.DIZZY
REPEAT 4 [ FD 30 RT 90 ] *
LT 45
H.C.DIZZY
END
```
*TRS—Use parentheses instead of brackets.

Session 19 Problem Sheet, page
APPLE, ATARI, MIT, TI

REPEAT	FD	RT
360	1	1
180	2	2
120	3	3
90	4	4
72	5	5
60	6	6
51.4	7	7
45	8	8
40	9	9
36	10	10

REPEAT	FD	RT
360	1	1
180	1	2
120	1	3
90	1	4
72	1	5
60	1	6
51.4	1	7
45	1	8
40	1	9
36	1	10

Session 19 Problem Sheet, page
TRS

REPEAT	FD	RT
360	1	1
180	2	2
120	3	3
90	4	4
72	5	5
60	6	6
51	7	7
45	8	8
40	9	9
36	10	10

REPEAT	FD	RT
120	3	3
90	3	4
72	3	5
60	3	6
51	3	7
45	3	8
40	3	9
36	3	10

Session 19 Challenger Sheet, page
APPLE, ATARI, MIT, TRS
(TRS—Use parentheses instead of brackets.)

1)
```
TO ONE.1
  REPEAT 360 [ FD 1 RT 1 ]
END


TO ONE.2
  REPEAT 180 [ FD 1 LT 2 ]
END
```

2)
```
TO TWO.1
  MOVE
REPEAT 90 [ FD 1 LT 4 ]
  REPEAT 120 [ FD 2 LT 3 ]
  REPEAT 180 [ FD 1 LT 2 ]
END


TO TWO.2
  MOVE
  HT
  REPEAT 90 [ FD 4 LT 4 ]
  REPEAT 90 [ FD 3 LT 4 ]
  REPEAT 90 [ FD 1 LT 4 ]
```

3)
```
TO THREE
  RT 45
  OVER
END


TO OVER
  REPEAT 90 [ FD 1 RT 1 ]
END
```

4)
```
TO FOUR
  RT 135
  UNDER
END


TO UNDER
  REPEAT 90 [ FD 1 LT 1 ]
END
```

5)
```
TO FIVE
  PU LT 90
  FD 60 RT 90
  PD RT 45
  OVER
  UNDER
END
```

6)
```
TO SIX
  RT 45
  OVER
  RT 90
  OVER
END
```

```
7) TO SEVEN
    LT 90
    SIX
    PU
    HOME
    RT 90
    FD 20
    LT 135
    PD
    SIX
    PU
    HOME
    LT 90
    FD 20
    LT 45
    PD
    SIX
  END


8) TO EIGHT
    FD 10 LT 90
    REPEAT 40 [ FD 1 RT 9 ]
    RT 90 BK 10
    LT 50
    REPEAT 3 [ LT 20 LFTLEG ]
    RT 160
    REPEAT 3 [ RT 20 RHTLEG ]
    HT
  END


  TO RHTLEG
    FD 50 RT 90 FD 30
    BK 30 LT 90 BK 50
  END


  TO LFTLEG
    FD 50 LT 90 FD 30
    BK 30 RT 90 BK 50
  END


9) TO NINE
  HT
  MOVE1
  HANDLEBARS
  MOVE2
  FRAME
  RWHEEL
  SEAT
END


TO MOVE1
  PU
  FD 60
  LT 90
  FD 20
  RT 180
  PD
END


TO HANDLEBARS
  FD 29
  RT 25
  FD 58
  RT 155
  FD 12
END


TO MOVE2
  BK 12
  LT 155
  BK 35
  RT 30
END


TO FRAME
  FD 95
  BK 30
  FWHEEL
  BK 65
  RT 80
  FD 110
END


TO FWHEEL
  RT 90
  REPEAT 180 [ FD 1 LT 2 ]
  LT 90
END


TO RWHEEL
  BK 30
  RT 90
  REPEAT 180 [ FD 1 LT 2 ]
  LT 90
END


TO SEAT
  BK 10
  RT 133
  FD 20
  LT 80
  REPEAT 10 [ FD 1 RT 1 ]
  RT 140
  REPEAT 10 [ FD 1 RT 1 ]
  RT 20
  REPEAT 10 [ FD 1 RT 1 ]
  RT 140
  REPEAT 10 [ FD 1 RT 1 ]
END
```

Session 19 Challenger Sheet, page TI

```
1) TO ONE.1
   PU FD 20
   PD
   REPEAT 360 [ FD 1 RT 1 ]
   END

   TO ONE.2
   REPEAT 90 [ FD 2 RT 4 ]
   END

2) TO TWO
   HT FD 20
   REPEAT 90 [ FD 4 LT 4 ]
   REPEAT 90 [ FD 3 LT 4 ]
   REPEAT 90 [ FD 1 LT 4 ]
   END

3) TO THREE
   RT 45
   OVER
   END

   TO OVER
   REPEAT 45 [ FD 1 RT 2 ]
   END

4) TO FOUR
   RT 135
   UNDER
   END

5) TO FIVE
   PU LT 90
   FD 60 RT 90
   PD RT 45
   OVER
   UNDER
   END

6) TO SIX
   RT 45
   OVER
   RT 90
   OVER
   END

7) TO SEVEN
   LT 90
   SIX
   PU
   HOME
   RT 90
   FD 20
   LT 135
   PD
   SIX
   PU
   HOME
   LT 90
   FD 20
   LT 45
   PD
   SIX
   END

8) TO EIGHT
   FD 10 LT 90
   REPEAT 40 [ FD 1 RT 9 ]
   RT 90 BK 10
   LT 50
   REPEAT 3 [ LT 20 LFTLEG ]
   RT 160
   REPEAT 3 [ RT 20 RHTLEG ]
   HT
   END

   TO RHTLEG
   FD 50 LT 90 FD 30
   BK 30 LT 90 BK 50
   END

   TO LFTLEG
   FD 50 LT 90 FD 30
   BK 30 RT 90 BK 50
   END

9) TO NINE
   PU FD 10
   PD
   HT
   MOVE1
   HANDLEBARS
   MOVE2
   FRAME
   RWHEEL
   SEAT
   END

   TO FRAME
   FD 95
   BK 30
   FWHEEL
   BK 65
   RT 80
   FD 110
   END
```

```
TO MOVE1
PU
FD 60
LT 90
FD 20
RT 180
PD
END
```

```
TO FWHEEL
RT 90
REPEAT 180 [ FD 1 LT 2 ]
LT 90
END
```

```
TO RWHEEL
BK 30
RT 90
REPEAT 180 [ FD 1 LT 2 ]
LT 90
END
```

```
TO HANDLEBARS
FD 28
RT 25
FD 58
RT 155
FD 12
END
```

```
TO MOVE2
BK 12
LT 155
BK 35
RT 30
END
```

```
TO SEAT
BK 10
RT 133
FD 20
LT 80
REPEAT 10 [ FD 1 RT 1 ]
RT 10
REPEAT 10 [ FD 1 RT 1 ]
RT 20
REPEAT 10 [ FD 1 RT 1 ]
RT 140
REPEAT 10 [ FD 1 RT 1 ]
END
```

NOTES

ICCE Organization Members

Alaska Association for Computers in Education (AACE)
Contact: Sandra Burgess
Coordinator, The Northern Institute, 650 W. International Airport Rd., Anchorage, AK 99502

Alberta Association for Educational Data Systems
Contact: Keath Austen
838 Education Tower, University of Calgary
2500 University Dr. NW., Calgary
Alberta, Canada T2N 1N4

Arizona Computer Users in Education
Contact: Nancy Watson
2148 Santos Drive
Tempe, AZ 85281

Association for the Development of Computer-Based Instructional Systems (ADCIS)
Contact: Dr. Gordon Haves, Executive Director
ADCIS—Miller Hall 409, Western Washington University, Bellingham, WA 98225

Computer Education Group of Victoria
Contact: Leon Guss, Conference Secretary
P.O. Box 245, Niddrie 3042
Melbourne, Victoria, Australia

Computer Education Group of Queensland
Contact: D.A. Carey
G.P.O. Box 1669
Brisbane, Queensland 4001
Australia

Computer Education Society of Ireland
Contact: Mary Devlin
85 Limetrees Road East, Douglas, Cork, Ireland

Computer Educators of Idaho
Contact: Dennis Dempsey
P.O. Box 1088, Hailey, ID 83333

Computers, Learners, Users, Educators Association (CLUES)
Contact: Henry J. Petersen, Executive Director
50 Nellis Drive, Wayne, NJ 07470

Computer-Using Educators (California)
Contact: Glenn Fisher
Alameda County Office of Education
School Computer Unit,
224 W. Winston Ave.,
Hayward, CA 94544-1198

Computer-Using Educators of British Columbia (CUEBC)
Contact: Barry Underwood
509 Alder St., Campbell River
B.C., Canada V9W 2N9

Computer-Using Educators of Kentucky (CUE-KY)
Contact: Janet Parker
EMCE Dept., School of Education,
University of Louisville, Louisville, KY 40292

DIDACOM
Contact: Inno Brookman
Avenbeeck 98, 2182 RZ Hillegom,
The Netherlands

Educational Computing Consortium of Ohio
Contact: Ellen Richman, Coordinator
4777 Farnhurst Road, Cleveland, OH 44124

Educational Computing Association of Western Australia
Contact: Mr. K. Penter
c/o Schools Computing, Education Dept.,
151 Royal St., East Perth, Western Australia

Educational Computing Organization of Ontario (ECOO)
Contact: Robert E. Drake, President
252 Bloor Street West, Toronto,
Ontario, Canada M5S 1V6

Educational Microcomputer Users Group of Central New York (EMUGCNY)
Contact: Jane McCrohan
Roxboro Middle School, Mattydale, NY 13211

Educators Interest Group of the San Diego Computer Society
Contact: Melvin L. Zeddies
P.O. Box 81537, San Diego, CA 92138

Florida Association of Science Teachers (FAST)
Contact: Richard Sweetsir
1147 Scotten Rd., Jacksonville, FL 32205

Illinois Association for Educational Data Systems (ILAEDS)
Contact: Lyle B. Smith, President
Computer Science Department, Northern Illinois University, Box 128, De Kalb, IL 60115

Indiana Computer Educators
Contact: David A. Flowers
Ft. Wayne Community Schools, Adm. Cwt.
1230 So. Clinton St., Ft. Wayne, IN 46802

Manitoba Association for Educational Data Systems (MAN-AEDS)
Contact: John Muller
27 O'Brien Crescent
Winnipeg, Manitoba Canada R3R 1M5

Michigan Association for Computer Users in Learning (MACUL)
Contact: Lary R. Smith, Communications Sect.
33500 Van Born Road, Wayne, MI 48184

Milwaukee Educational Computing Association
Contact: Henry Kepner, Jr.
Dept. of Curriculum & Instruction
University of Wisconsin—Milwaukee
P.O. Box 413, Milwaukee, WI 53201

Minnesota Association for Educational Data Systems
Contact: Bill Bjorum
1875 S. Greeley, Stillman, MN 55082

Montana Council for Computers in Education
Contact: Leon Roland
1846 Bitteroot
Billings, MT 59105

National Institute for Microcomputer Based Learning (NIMBL)
Contact: Stanley Silverman, President
348 Plymouth Ave., Brightwaters, NY 11718

New Hampshire Association for Computer Education Statewide (NHACES)
Contact: Anne Knight, President
University of New Hampshire, Computer Services, Durham, NH 03824

New Mexico Council of Computer Users in Education
Contact: Dr. Phil H. Barck, President
P.O. Drawer 90, Los Alamos, NM 87544

New York State Association for Educational Data Systems
Contact: Don Ross, President
Ardsley High School, 300 Farm Road,
Ardsley, NY 10502

New South Wales Computer Education Group
Contact: Toni Downes, Chairperson
P.O. Box 148, Broadway, N.S.W. 2007 Australia

Northwest Council for Computer Education (Oregon, Washington and Northern Idaho)
Contact: Howard Bailey
Computing Center, Eastern Oregon State College
La Grande, OR 97850

Oklahoma Educational Computer Users Program (OECUP)
Contact: Richard V. Andree
Department of Mathematics, 601 Elm St., #423,
The University of Oklahoma, Norman, OK 73019

Pennsylvania Learning Resources Association (PLRA)
Contact: George H. Zook, Director, IMS
Lancaster-Lebanon I.U./IMS
P.O. Box 5026, Lancaster, PA 17601

Quebec Association of Computer Users in Education
Contact: Don Rosenbaum, Membership Director
P.O. Box 369, Ormstown, Quebec, J0S 1K0,
Canada

Saskatchewan Association for Computers in Education
Contact: Duncan Campbell
Mt. Royal Collegiate, 2220 Rusholme Road,
Saskatoon, Saskatchewan, Canada S7L 4A4

The Science Teachers' Association of Ontario
Contact: Warren Sirrs
1032 Harkness Ave., Ottawa, Ontario,
Canada K1V 6P2

Society of Data Educators (SDE)
Contact: Robert Behling
Bryant College, Smithfield, RI 02917

Society for Microcomputers in Life & Education
Contact: Hillel Weintraub, Acting President
Doshisha International High School
Tatara, Tanabe-cho, Tsuzuki-gun
Kyoto-fu, Japan 610-03

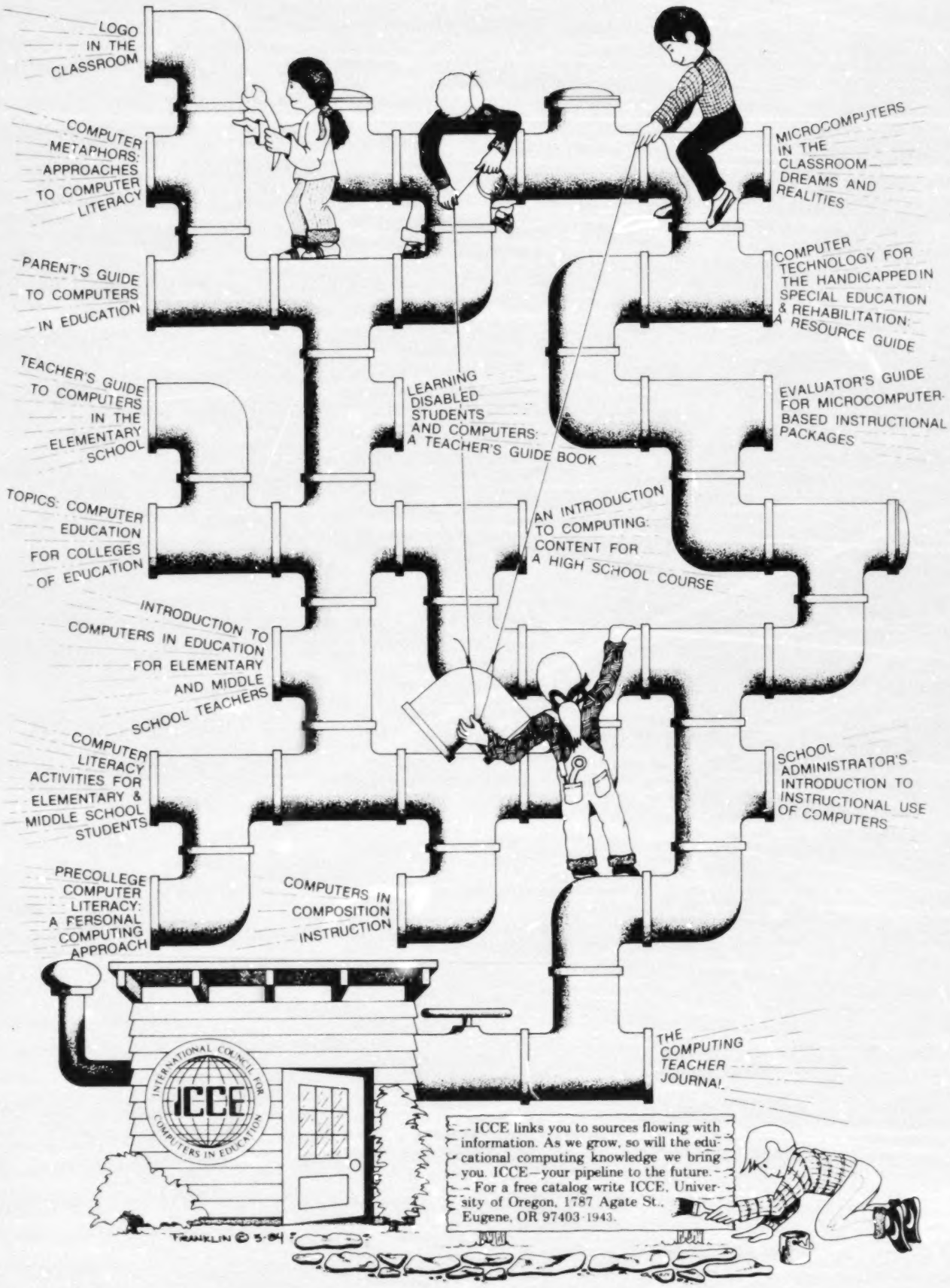
Texas Computer Education Association
Contact: Vicki S. Smith
P.O. Box 2573, Austin, TX 78768

The Utah Council for Computers in Education
Contact: Dr. Larry C. Christensen
1295 North 1200 West, Mapleton, UT 84663

Wyoming Educational Computing Council
Contact: Robert L. Morissette, President
Laramie County School District #1, Admin. Bldg.
2810 House Ave., Cheyenne, WY 82001

Young People's LOGO Association (YPLA)
Contact: James H. Muller
1208 Hillsdale Drive, Richardson, TX 75081

BEST COPY AVAILABLE


LOGO IN THE CLASSROOM
COMPUTER METAPHORS: APPROACHES TO COMPUTER LITERACY
PARENT'S GUIDE TO COMPUTERS IN EDUCATION
TEACHER'S GUIDE TO COMPUTERS IN THE ELEMENTARY SCHOOL
TOPICS: COMPUTER EDUCATION FOR COLLEGES OF EDUCATION
INTRODUCTION TO COMPUTERS IN EDUCATION FOR ELEMENTARY AND MIDDLE SCHOOL TEACHERS
COMPUTER LITERACY ACTIVITIES FOR ELEMENTARY & MIDDLE SCHOOL STUDENTS
PRECOLLEGE COMPUTER LITERACY: A PERSONAL COMPUTING APPROACH
COMPUTERS IN COMPOSITION INSTRUCTION
LEARNING DISABLED STUDENTS AND COMPUTERS: A TEACHER'S GUIDE BOOK
AN INTRODUCTION TO COMPUTING: CONTENT FOR A HIGH SCHOOL COURSE
MICROCOMPUTERS IN THE CLASSROOM— DREAMS AND REALITIES
COMPUTER TECHNOLOGY FOR THE HANDICAPPED IN SPECIAL EDUCATION & REHABILITATION: A RESOURCE GUIDE
EVALUATOR'S GUIDE FOR MICROCOMPUTER-BASED INSTRUCTIONAL PACKAGES
SCHOOL ADMINISTRATOR'S INTRODUCTION TO INSTRUCTIONAL USE OF COMPUTERS
THE COMPUTING TEACHER JOURNAL
INTERNATIONAL COUNCIL FOR COMPUTERS IN EDUCATION
ICCE
— ICCE links you to sources flowing with information. As we grow, so will the educational computing knowledge we bring you. ICCE—your pipeline to the future. — For a free catalog write ICCE, University of Oregon, 1787 Agate St., Eugene, OR 97403-1943.
FRANKLIN © 5-84

BEST COPY AVAILABLE